

CAD - CAM INTEGRATION OPTIMIZING DATA TRANSFER BETWEEN DIFFERENT INTEGRATED DESIGN SYSTEMS IN VIRTUAL ENTERPRISE ENVIRONMENT

Carmen Cristiana BUCUR¹, Traian AURITE²

Rezultatele cercetării publicate în această lucrare se bazează pe un studiu de caz pentru proiectarea arborilor folosiți la mașinile unelte, utilizând patru Sisteme Integrate de Proiectare. Am început cu realizarea unui exemplu de diagrame SADT utilizate în transferul de date și industria CAD/CAM, pe baza lor am construit arhitectura UML. Transpunând arhitectura UML în limbajul de programare C++ obținem agentul sursă care poate fi transformat într-un agent de import/export de către un programator cu scopul de a îmbunătăți transferul de date într-o Întreprindere Virtuală. În aceste condiții utilizatorii ar putea importa piesele cu istoric de la un IDS la altul.

The research results published in this paper are based on a case study for shaft design made for machine tools using four Integrated Design Systems. At the beginning we elaborate the SADT diagram used in CAD/CAM Industry and data transfer. Based on these diagrams we build UML architecture. Transposing the UML in C++ Programming language in order to improve the data transfer in a Virtual Enterprise we obtain the agent source which can be transformed in an import/export agent by a programmer, then the users could import the part with historic from one IDS to another.

Keywords: data transfer, CAD/CAM, UML, ISO 10303/STEP, C++

List of Symbols

CAD – Computer Aided Design
CAM – Computer Aided Manufacturing
CAE – Computer Aided Engineering
CNC – Computer Numerical Control
IDS – Integrated Design System
ISO – International Organization for Standardization
STEP – ISO 10303 - Standard for the Representation and Exchange of Product Data
SADT - Structured Analysis and Design Technique
UML – Unified Modeling Language

¹ PhD student, Faculty of Engineering and Management of Technological Systems, University POLITEHNICA of Bucharest, Romania, e-mail: cristiana.bucur@gmail.com

² Prof., Faculty of Engineering and Management of Technological Systems, University POLITEHNICA of Bucharest, Romania, e-mail: aurite@imst.msp.pub.ro; aurite@amcsit.ro

1. Introduction

The CAD/CAM systems in existence today have very limited capability to allow the users to perform conceptual designs. This is because most of the existing CAD/CAM systems require exact geometric specifications including shapes and dimensions, even though these details are unnecessary and often unknown at the conceptual design stage.

From at least 10 years the researchers have been studying the problem on CAD/CAM integration and trying to optimize data transfer between different IDS Systems.

Modern manufacturing enterprises are built from facilities spread around the globe, which contain equipment from hundreds of different manufacturers. Immense volumes of product information must be transferred between the various facilities and machines. Today's digital communications standards have solved the problem of reliably transferring information across global networks. For mechanical parts, the description of product data has been standardized by ISO 10303 (STEP). This leads to the possibility of using standard data throughout the entire process chain in the manufacturing enterprise. Barriers to realizing this principle are the data formats used at the machine level. Most computer numerical control (CNC) machines are programmed in the ISO 6983 "G-code" language. Programs are typically generated by computer-aided manufacturing (CAM) systems that use computer-aided design (CAD) information. However, "G-code" limits program portability for three reasons. First, the language focuses on programming the tool centre path with respect to machine axes, rather than the machining process with respect to the part. Second, the standard defines the syntax of program statements, but in most cases leaves the semantics ambiguous. Third, vendors usually supplement the language with extensions that are not covered in the limited scope of "G-code" [1].

In our research we discovered barriers in product data transfer produced by: different users operating in a different manner Integrated Design Systems (IDS); number of different organizations involved in the definition and manufacture of products, each having their own distinct set of rules; the development of the second IDS, which used different data to represent the same information as the first system, providing some new functions and types of information; the information created in one system cannot be read and processed by the other system; from the lifecycle of many products, which extends far beyond the effective working life of one particular IDS and its hardware; product support must extend until the last product is scrapped [2].

The options discovered until now to solve these problems are: the simplest option would be to enforce the use of the same IDS system throughout a project; the second approach is to recognize the need for multiple IDS, and to build

translators between the various systems; the third approach is to develop a neutral form of data with which each system can communicate [2].

One problem is the discordance which appears after the data transfer between one Integrated Design System to another, and the biggest problem is that the part can't be edited anymore [3].

Solving some problems which appear in virtual enterprise from industrial organizations using different IDS from one department to another is important for the industrial partners involved in such architecture as well as for the software developers.

In present are built numerous translators between the various systems but none of them are able to import the parts with historic. The main focus of this study is to create the UML architectures for the CATIA and SOLID EDGE Software in order to improve the data transfer in a Virtual Enterprise. Solving the problems which appear using different IDS from one department to another is important especially to reduce design and manufacturing time, errors and even blanks rebooting.

2. Case Study

Based on the main issue presented, we elaborate an SADT frame of an IDS data transfer model in order to improve CAD/CAM communication. We develop this SADT until the "Generating the 2D or 3D parts/assemblies in CAD Module" activity. There are two ways to elaborate the parts/assemblies in IDS: the first one is to design the parts/assemblies directly in actual IDS and the second is to import the parts/assemblies model built in previous IDS [4, 5].

The Structured Analysis and Design Technique (SADT) is a graphical representation that can be transposing in a programming language in order to improve CAD/CAM communication between different IDS using STEP format and to elaborate a CAD/CAM interface.

After analyzing all the possibilities of improving the transfer with historic between different IDS we considered that the proper solution to be studied further is to intermediate the transfer with an agent. This agent is supposed to read all the steps made by the designer and translate them in the destination IDS each at a time. To solve this problem we need an intelligent agent that we started to develop in UML 2.0. The model of agent source was structured only for shafts. Based on this framework anyone is able to expand it for specific parts or other IDS.

At the beginning we defined the main objects: shaft and pocket, in four diagrams named CATIA, SOLID EDGE, SOLID WORKS and UNIGRAPHICS. The diagrams must have similarly architecture in order to relate the objects in each of them.

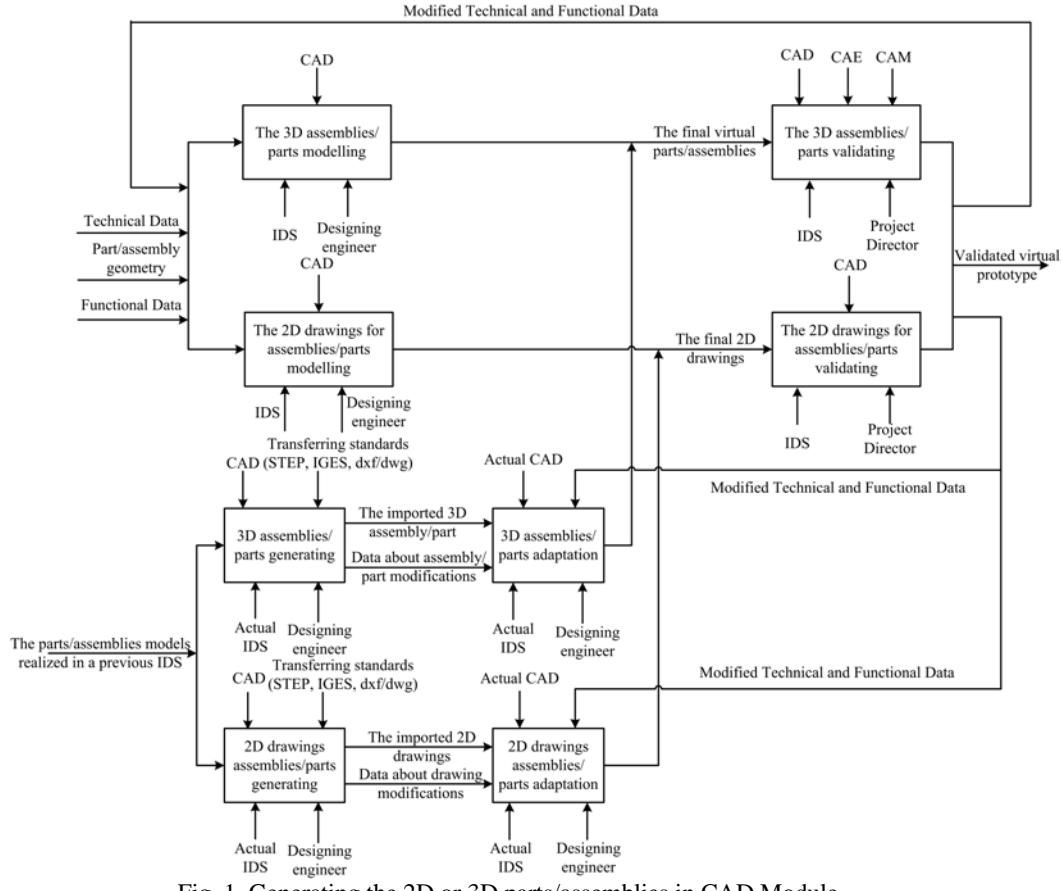


Fig. 1. Generating the 2D or 3D parts/assemblies in CAD Module

CATIA diagram presented in Fig. 2, incorporate the *PART_name* Class which import the *Shaft* and *Pocket* commands. In *PART_name* Class is defined the *part_name* attribute and operation as String type, witch shall import the real name of the part. The *Shaft* Class defines the *shaft_no* attribute, in order to import them in chronological order and operation as Integer type and is associated with *Angle*, *Axis* and *Sketch* options. The *Angle* Class is characterized by *Value* attribute and operation defined as Integer type, to import/export the degrees number. *Axis* Class appeals the *3D Line* that uses *3D Point* Class. *Axis_no* parameter is described as Boolean type. The *Pocket* option is associated with *Sketch* and *Height* Class. *Pocket_no* attribute and operation is described as Integer type. Each of these classes is importing/exporting the values, the coordinates, the elements that are used by the command from the other classes that are reading them from the original model in order to write them in the new one.

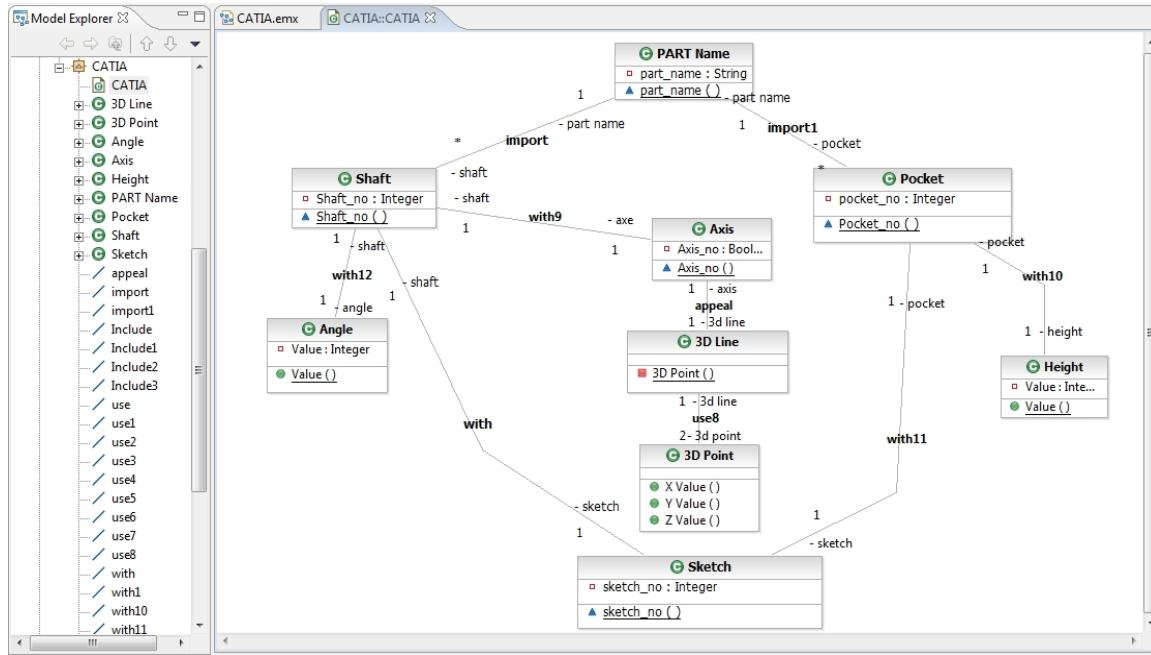


Fig. 2. The UML architectures for shafts designed for CATIA made in IBM Rational Software Architect

In Fig. 3 is shown the diagram that we design for the *Sketch*. This diagram contains the following classes: *Absolute Axis*, *Constrains* and *Geometry*. The *Constrains* Class includes the next attributes: *Coincidence*, *Parallelism*, *Tangency*, *Offset*, *Perpendicularity*, *Angle*, *Radius* and *Length*. The *Geometry* Class includes the *Point*, *Line* and *Circle* attribute and options. All of these are supposed to take from the original part sketch the information like: sketch plan coordinates, the axis directions, the geometry, the value attributed to each object, constrains between witch objects and value if necessary, to export them to another IDS.

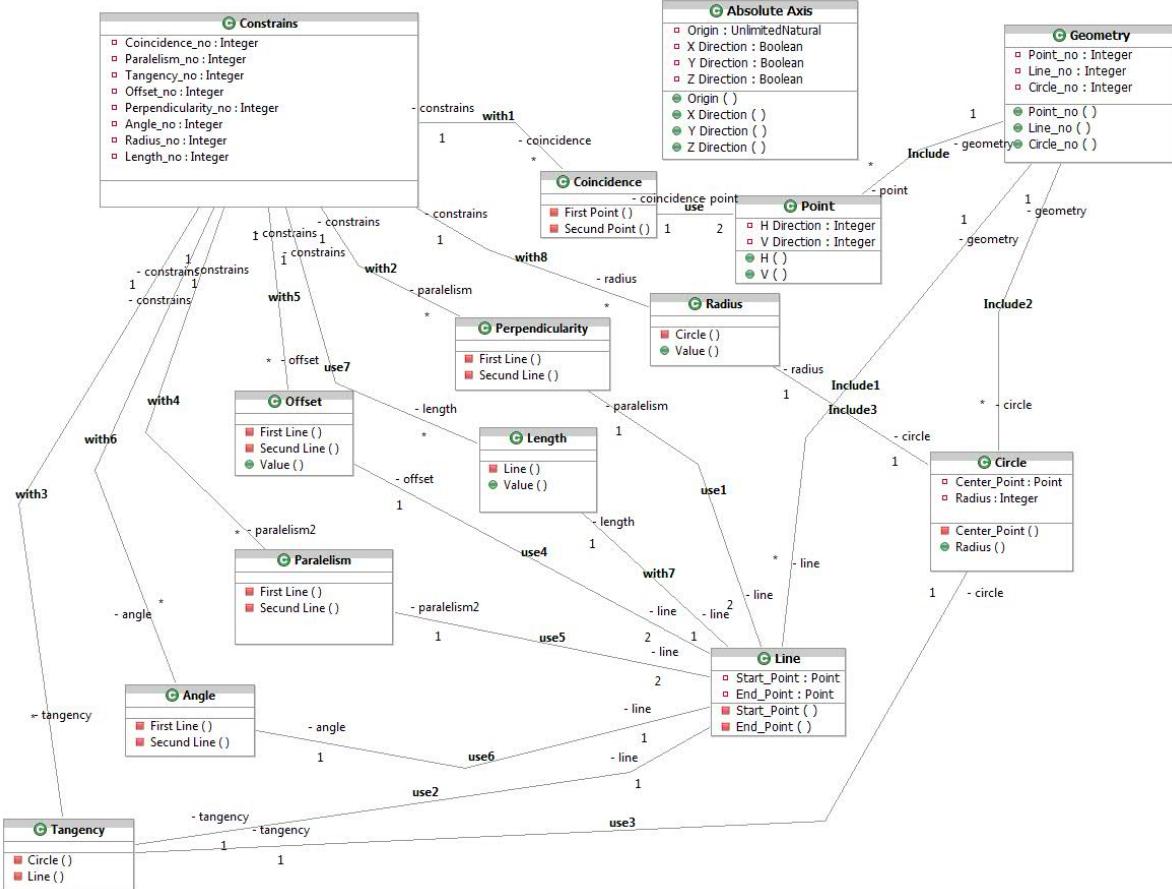


Fig. 3. The Sketch Content from UML architectures

SOLID EDGE diagram exposed in Fig. 4, also incorporate the *PART_name* Class which import the *Revolved Protusion* and *Cutout* commands. The *Revolved Protusion* Class is associated with *Angle*, *Axis of Revolution* and *Profile* options. The *Cutout* Class is associated with *Profile* and *Height* Class. The attribute and operation type defined in class diagrams are the same like in CATIA diagram to make the import/export operations of the attribute in each object functionally between the four IDS considered. This way we will be able to relate the *shaft_no* from the CATIA to *Revolved Protusion* from this diagram and all the other items.

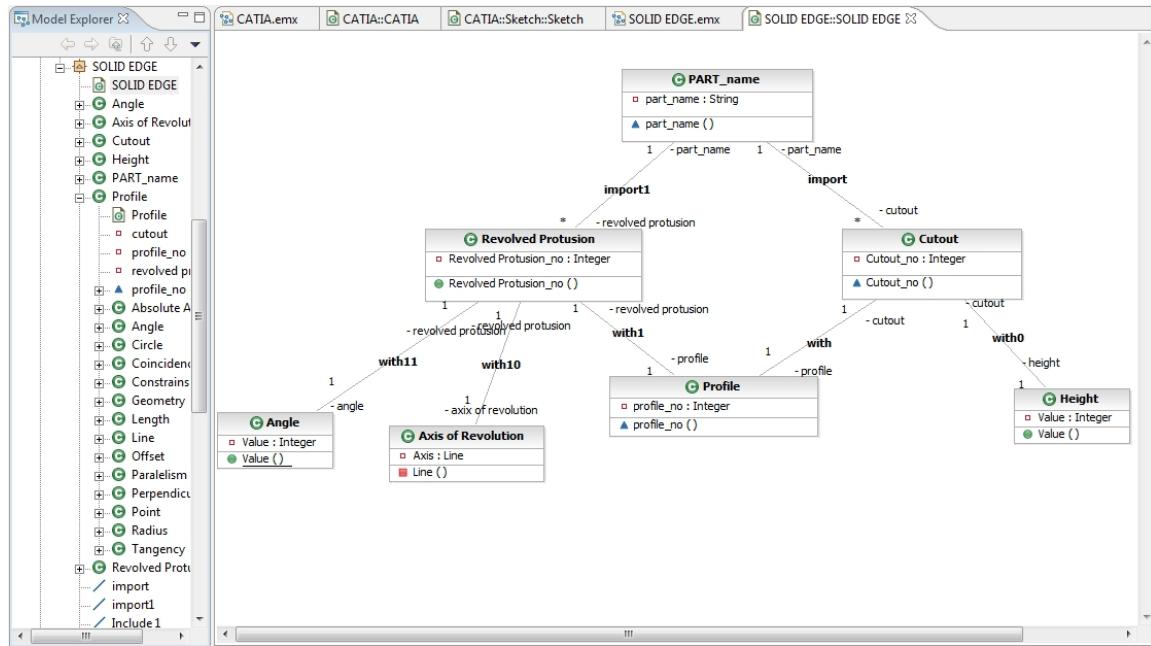


Fig. 4. The UML architectures for shafts designed for SOLID EDGE made in IBM Rational Software Architect

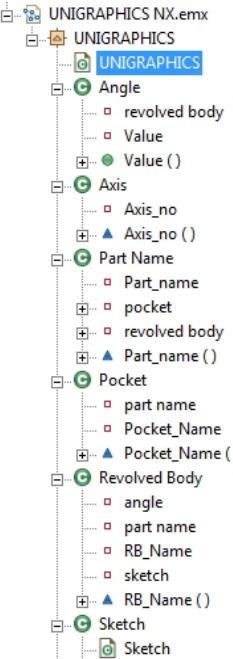


Fig. 5. a) The UML architectures for sketch and the diagram tree designed for UNIGRAPHICS made in IBM Rational Software Architect

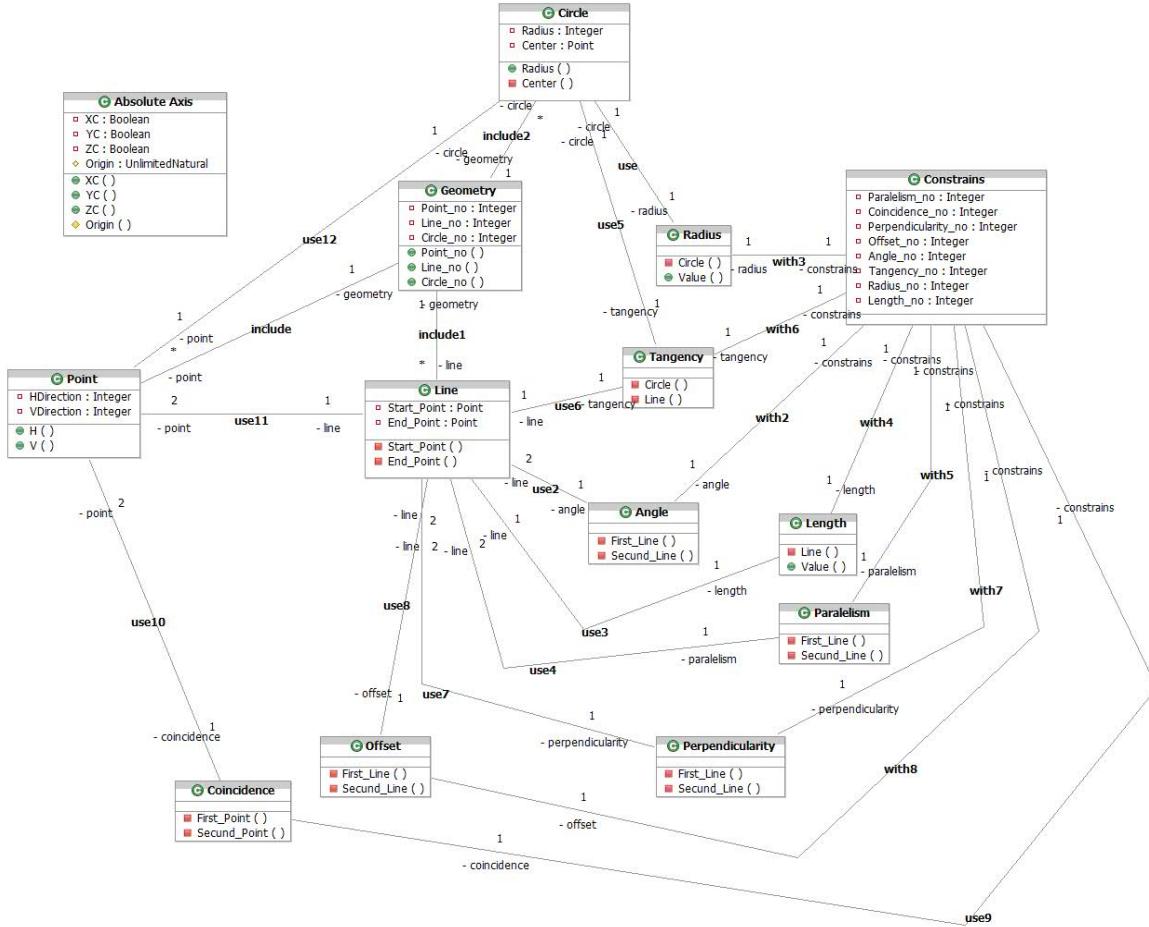


Fig. 5. b) The UML architectures for sketch and the diagram tree designed for UNIGRAPHICS made in IBM Rational Software Architect

For UNIGRAPHICS (Fig. 5) and SOLID WORKS (Fig. 6) we designed similar architectures like for CATIA and SOLID EDGE, for the same compatibility issue.

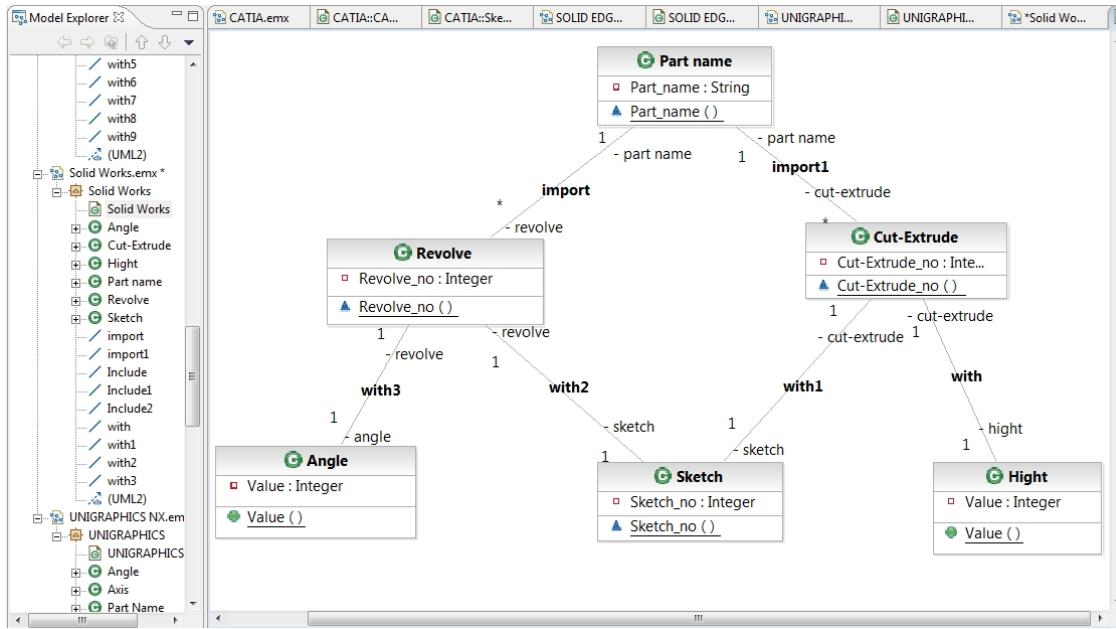


Fig. 6. The UML architectures for shafts designed for SOLID WORKS made in IBM Rational Software Architect

Transposing the UML in C++ Programming language, we obtain the agent source that can be transformed by a programmer in an import/export agent. This agent can be developed in software that will be able to import and export parts with full historic, between CAD software. The logical scheme of this source is presented in Fig. 7.

Fig. 8 is presenting the final stage of our project, the validation of the C++ fragments that define each class of the UML diagrams.

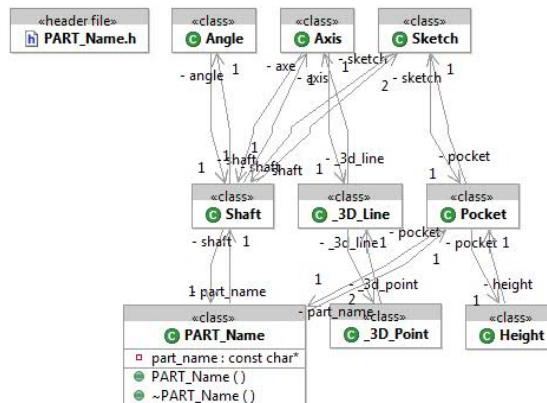
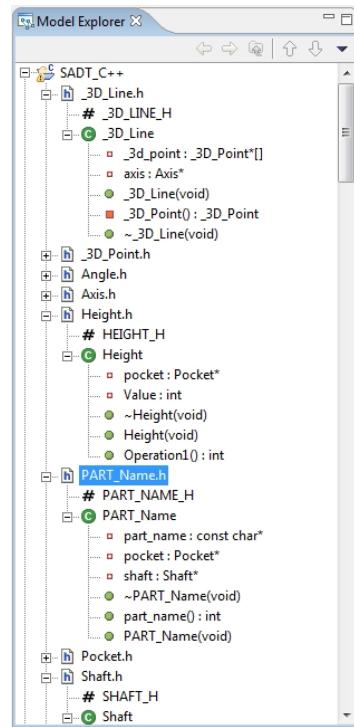


Fig. 7. The logical scheme of the code

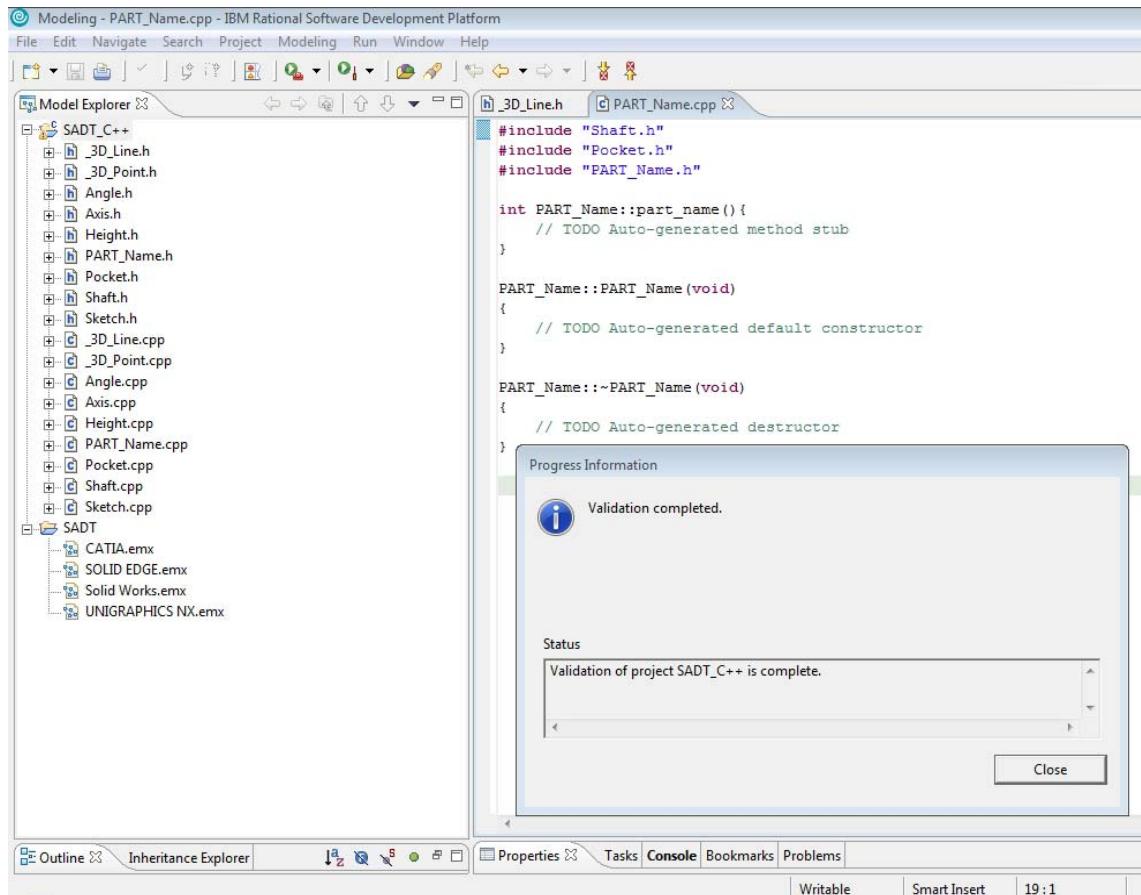


Fig. 8. A sequence of C++ Programming language

5. Conclusion

The biggest problem in virtual industries is the need to work with the same IDS, in order to keep the historic of each piece of an assembly. This is nowadays one of the criteria of choosing the partners. If the IDS used by the designer is different by the one used by the programmer, the last one will use an imported model with no historic. If there is a revision of the model made by the designer, all the programs for the machining will be lost, just because the model used has no historic and it is like a read only model. In most of the companies there are lots of stories about this transfer issue, all of them with time, money lose.

The compatibility of CAD/CAM/CAE is similar to that used to justify any technology-based improvement in manufacturing. It grows out of a need to continually improve productivity, quality and competitiveness. There are also other reasons why a company might make a conversion from manual processes to

CAD/CAM: increased productivity, common database with manufacturing, better quality, better communication between posts, common database with manufacturing, reduced prototype construction costs, faster response to clients/partners.

All these issues would disappear if the parts/assembly transfers between different IDS with historic would be possible. This is the reason why a transfer solution is searched for so many people.

Starting from the similarity of design commands in any IDS, *Shaft*, *Revolved Protusion*, *Revolve*, *Revolved Body* and *Pocket*, *Cut Extrude*, *Cutout*, we created the UML 2.0 architectures for shafts designed in CATIA, SOLID EDGE, UNIGRAPHICS and SOLID WORKS in order to transpose the UML in C++ Programming language and to obtain the agent source which can be transformed in import/export software by a programmer. After that the users of different IDS will be able to work on the part with full historic.

The main idea of this architecture is to be the start point of the future professional software that will improve CAD/CAM/CAE communication between any software.

This software is going to be able to read the steps made by the designer in the IDS that he is using and to actually redesign the part in the destination IDS, making the same moves like the designer, but using the other IDS commands. This way the part will be editable after each transfer.

R E F E R E N C E S

- [1] *J. Sääski, T. Salonen, J. Paro*, “Integration of CAD, CAM and NC with StepNC”, in VTT Working Papers, no. 28, ISBN 951.38.6580.0 (URL: <http://www.vtt.fi/inf/pdf/>), June 2005, 23 p
- [2] *José Barata*, “Introduction to Modelling and STEP”, Available at: <http://www.uninova.pt/~escn/storage/U2PIUninova.pdf>, Accesed on 2009-03-18
- [3] *C.C. Constantinescu (Bucur), L.F. Ciobanu, C.L. Popa*, “Study about the Manufacturing Integration in the Design Phase Using Several CAD-CAM Integration Systems Software”, Proceedings of The 1st International Conference “From Scientific Computing to Computational Engineering” IC-SCCE 2004, Tsahalis, T. (Ed.), ISBN 960-530-071-0, Greece, September 2004, Patras University Press, Athena, pp. 1195-1202
- [4] *C.C. Bucur*, “SADT Frame of an IDS Data Transfer Model Using STEP Format”, *Proceedings of The 16th International DAAAM Symposium*, Katalinic, B. (Ed.), ISSN 1726-9679, Austria, Published by DAAAM International, Vienna, October 2005, pp. 47-48
- [5] *C.C. Bucur, L.F. Ciobanu*, “Data Transfer and Simulation for Virtual Enterprise Environment Integration” Proceedings of The 7th Conference on Management of Innovative Technologies. MIT 2004, Chircor, M. & Dragoi, G. (Ed.), ISBN 973-700-028-5, Romania, AIUS, Constanta, October 2004, pp. 19-22