Integration of Furniture Design /Evaluation Tools
A STEP/Express Approach

Ricardo Jardim- Gonçalves, Manuel Barata, Adolfo Sanches Steiger Garçao
Intelligent Robotics Center
UNINOVA
Quinta da Torre - 2825 Monte da Caparica
PORTUGAL

Marcelo Tavares
Universidade Federal de Santa Catarina
88040-90 Florianopolis
BRAZIL

Abstract: The integration of software tools into CIM environment is discussed under standardization of information modelling and communication point of view. A STEP (ISO10303) standard based platform for integration of applications, developed for the European BRITE/EURAM CIMTOFI project, is described. The integration of CAD systems and Furniture Design / Evaluation into this platform is also presented. The communication protocol, available within the platform, is described. The current status of the work is presented, and the tendencies of evolution of the work are identified. The formation of an ISO Working Group on the definition of a Furniture Design Application Protocol is also discussed.

Keywords: Integration, Flexible Manufacturing, Information System, Information Modelling, STEP, CAD/CAM, Expert System.

1. Introduction

The integration of tools into CIM environments field has been intensively studied by the international scientific and industrial communities. Several questions have been put concerning implementation strategies.

The scenery actually found in the computer integrated manufacturing world, is the one where a large amount of tools has been developed without a global view of the manufacturing system. It gave rise to an effect we could call "Babel Tower", characterized by the fact that the data shared by several tools are modelled in different ways by each tool. So, data communication between tools is an arduous task to carry out.

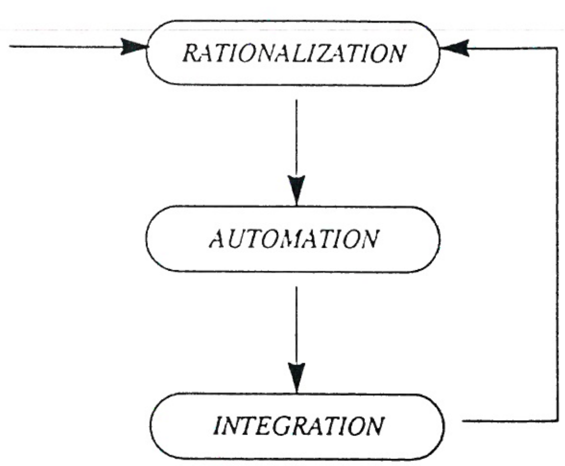
A pleasant situation would be the one in which all tools shared the same data structure, eliminating interface needs between them. Of course this is not a real situation, and ,to a certain point, it proves undesirable because the data structure can affect positively the performance of some tools and, simultaneously, negatively the performance of some others. In fact, what we have is a "salad" of data models, access methods and data storing technologies, that the tools' developers must assimilate.

According to Charles Edquist and Staffan Jacobson, "CIM means the integration of the whole process from the receipt of orders through design, business planning, purchasing (of components and raw materials), machining, inventory control (of parts, materials and finished products), automated warehouses, automatic vehicles, assembly, packing and marketing... CIM is still mainly theory and conjecture, a vision of the future" [1].

This statement tells us that the question of the integration spreads out on different fronts. The training of people involved in its conception and development, the economic interests of the CIM tools' suppliers, external obstacles between them and the consumer enterprises, and even internal obstacles between several areas to be integrated are examples.

The paper treats this question from a technical point of view, highlighting the effects caused by the

- Gradual Integration



adoption of standards in information modelling and communication. A proposal for the resolution of CIM integration problem is presented.

2. Integration

A brief "trip" around various words related to the integration term: union, fusion, unity, coherence, connection, jointure, affinity, relationship, association, federation, co-operation, cohesion, alliance, collectivism, sharing, among others, gives us a good hint at how extensive this theme is, and at how many implications it can have on the scene of the manufacturing world.

So, we are confronted with the challenge of condensing several tools into one global system, perfectly united, cohesive, coherent, interrelated, co-operative. These tools share information and constitute a chain that comprehends the whole life cycle of a product, since conception, production and marketing.

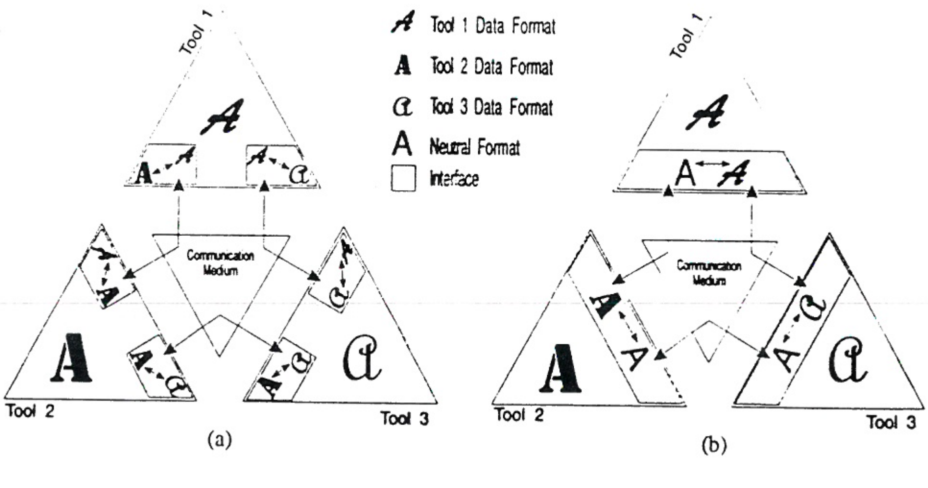
The integration task has a high degree of difficulty, so the adoption of a strategy, that is intrinsically related to the rationalization and automation of the activities involved in the manufacture, is needed.

The large amount of activities to be considered leads to a gradual, step by step integration strategy, in which several activities are carefully rationalized, automated, and subsequently inserted in the global system. Figure 1 depicts the gradual integration cycle.

In the chapter of the implementation, one of the most important parts is played by the interfaces. These elements allow the information sharing and make the tools' interoperability feasible. Beyond the physical access, the interfaces must have capacity of information encoding and decoding, according to pre-defined formats.

Historically, the first steps towards industrial automation were made through localized initiatives of some enterprise sections. The formation of island of automation led to that that the normal integration approach was pairwise. In spite of the benefits obtained through the referred islands, this initial phase was worriless with the global integration of

Interfaces between tools



the activities involved in the manufacturing process.

This kind of "hard" approaches requires a high number of interfaces ((n2 - n) in the extreme case of integration of n tools among each other). As the use of interface is crucial for the tool integration and their developments have high costs, the researches have been driven toward "*flexible*" approaches, in particular those leading to the utilization of n interfaces, at most. Examples of these kinds of approaches are the integration platform CIM-BYOSYS [8] and the series of standards ISO10303 - STEP [3]. The next Figure depicts the pairwise integration (a) and the use of flexible interfaces (b).

These researches proved the fact that a perfect and gradual integration of the manufacturing tools, depends on three essential factors, related in [2]:

i) Definition of common models for shared concepts, in order to support an effective exchange of information. A circle in 2D space, for example, could be modeled in many ways: through a point and the radius, or through three points, or even through the coefficients of its equation, among others.

ii) Adoption of an Information Management System.

iii) Realization of an integrating infrastructure that provides a functional support for integration, i.e., a kind of "software bus" offering high level inter-process communication services to the connected tools.

That is to say that all tools must be able to model the information according to only one formalism. Besides, tools should be able to share data with several other tools, using only one interface.

Thus, a secure integration of the activities involved in the product life cycle can be achieved. The security needs be guaranteed since the initiative of standardization affects the crucial phases of the information's treatment, as modelling and communication interfaces.

3. Standardization

The first steps to the definition of a standard for modelling and communication of information in manufacturing environments had its origin in the late 70's, with the introduction of drafting tools in the design areas of enterprises. The necessity for changing drafts between systems of various suppliers became apparent when using *plot-files.* Standard formats, to represent graphic entities, were defined. However, these *plot-files* were limited in their capability of rigorously representing drafts, what made increase the need for a neutral specification of the interchange between tools.

In the 80's several organizations published standards for the representation of the information produced by CAD system. IGES, VDA-FS and SET are examples of such standards.

The increasing number of standards and the need for representing information about products (besides their geometrical representations) brought in the preoccupation with the creation of a unique international standard. In 1984, the ISO created the sub-committee SC4, with the following objective: "...to specify a form for the unambiguous representation and exchange of computer interpretable product information throughout the life of a product..." [3]. This standard was called ISO 10303, or STEP (*STandard for the Exchange of Product model data*).

The STEP standard includes a formal language for information modelling, denominated EXPRESS [ISO02]. One of the characteristics of the EXPRESS language is a mechanism, the schema, that allows the definition of information models for whichever areas of application, be it design, process planning, marketing, and others.

-EXPRESS [1] *Schema*

```
SCHEMA geometry_2D;
ENTITY length_measure;
        amount:REAL;
        unity;ENUMERATION OF
(                       meter,inch);
WHERE
        WR1: amount>0;
END_ENTITY;(*length_measure*)

ENTITY geometric_element
        ABSTRACT
        SUPERTYPE OF (point,line,circle);
        id: STRING;
END_ENTITY;
```

[1] *value* is an example of attribute and the WR1 clause is an example of constraint.

```
ENTITY point
        SUBTYPE OF (geometric_element);
        x: length_measure;
        y: length_measure;
END _ENTITY; (*ponto*)

ENTITY line
        SUBTYPE OF (geometric_element);
        start:point;
        end:point;
END_ENTITY;(*line*)

ENTITY circle
        SUBTYPE OF (geometric_element);
        centre:point;
        radius: length_measure;
END_ENTITY;(*circle*)

END_SCHEMA; (*geometry_2D*)
```

Each schema contains the specification of entities, i.e. objects of interest in a specific domain. The entities are defined in terms of attributes, that can in their turn be constrained. The EXPRESS language implements the concept of inheritance, by means of an entity which can be defined as a specialization of others. Similarly, the information can be modelled in a taxonomic way. The Figure depicts a simplified schema of the 2D geometry domain.

To physically represent the data to be exchanged, a neutral file format was specified. A mapping between EXPRESS definition of an entity and the physical representation of its instances in the exchange file was also defined [5]. Figure 4 depicts the exchange file with instances of entities of the geometry-2D schema .

Several forms of implementation of data exchange between tools were approached in the elaboration of the STEP standard:

i) **Physical File**: exchange of product data via files. The files are processed in a fixed order (reading and writing from the start to the end of the file);

ii) **Working Form:** this implementation form is based on the exchange of files, but the data can be accessed by a computer application at random through its storage in core memory;

iii) **Data Base**: the product can be shared between computer application through a common database.

iv) **Knowledge Base**: Same as iii, but this implementation form allows for the checking of rules and constraints on the data, not present in the other implementation forms (i, ii and iii).

4. Integration Platform

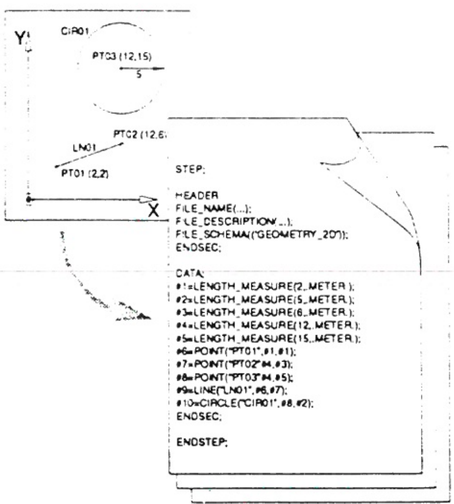
This section describes a STEP standard based integration platform. The implementation approach iv, depicted in Figure 5, was adopted. The platform is under development in the Intelligent Robotics Group, within the ambit of the BRITE CIMTOFI project.

The integration platform, depicted in Figure 6, is fundamentally constituted by an EXPRESS Compiler and an Information Management System (IS).

EXPRESS Compiler

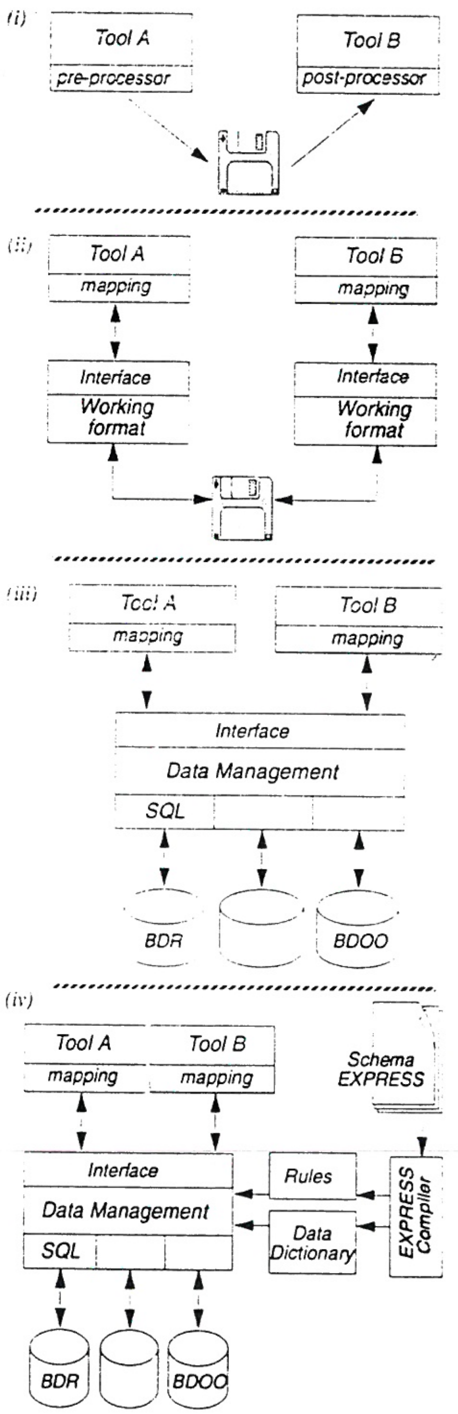
The validation of information accomplished by the IS, depends on a model (*schema* EXPRESS) elaborated previously. This model, also called meta-information, must be introduced in the meta-information dictionary .This task falls under the responsibility of the EXPRESS compiler, that processes the EXPRESS schema, generating the meta-information representation structure and its transfer to the meta-information dictionary.

Mapping for the neutral format



Data exchange implementations philosophies defined in STEP standard

- *Data exchange implementations philosofies defined in STEP standard.*



Information System

This is the nucleus of the integration platform. It plays the part of an information server , giving access to the data produced and consumed by external systems (EXPRESS compiler, tools and data persistent storage system).

The following elements will make up an information system:

- **Communication Protocol**: Library of functions , provided the capability of local or remote communications between the IS and the external systems. The protocol has also the function of validating the transmitted data, based on the meta-information dictionary. The protocol is divided into:

  ISAP - Information System Access Protocol: Set of primitives with the capability of managing, manipulating and accessing instances of EXPRESS entities.

  MISAP - Meta ISAP: Set of primitives that allows the manipulation of meta-information. The primitives of ISAP are used in its implementation (see Figure).

  PISAP - Persistent ISAP: Set of primitives that allows data transfer to a persistent storage medium. It also allows the transfer of meta-information to the persistent medium, avoiding the recompilation of schemata, whenever required. The primitives of ISAP are used in its implementation .

- **MID - Meta-information Dictionary**: Storage structure of which contents is provided by the EXPRESS compiler.

- **PDB - Product Data Base**: Storage structure of the data proceeded from the tools.

As an implementation strategy, all the information transmitted between the IS and the external systems, is modelled by EXPRESS schemata. Consequently, the transmitted data are always instances of EXPRESS entities, making it possible to use the communication protocol itself in the transmission and manipulation of all kind of information (even meta-information, provided that the users are getting access to it).

To this effect there are considered three levels of information support structures.

- Incorporated in the IS code is defined the support structure of the **meta-information model**. This structure allows the use of a communication protocol for updating, in the IS, the models that define the tools' schemata.
- The second support structure level results from the compilation of the **meta-information model**. This structure allows the use of a communication protocol for updating, in the IS, the tools' schemata.
- Finally, the third support structure level results from the compilation of the tools' schemata, or simply **meta-information**. This structure allows the use of a communication protocol to manipulate the product data.

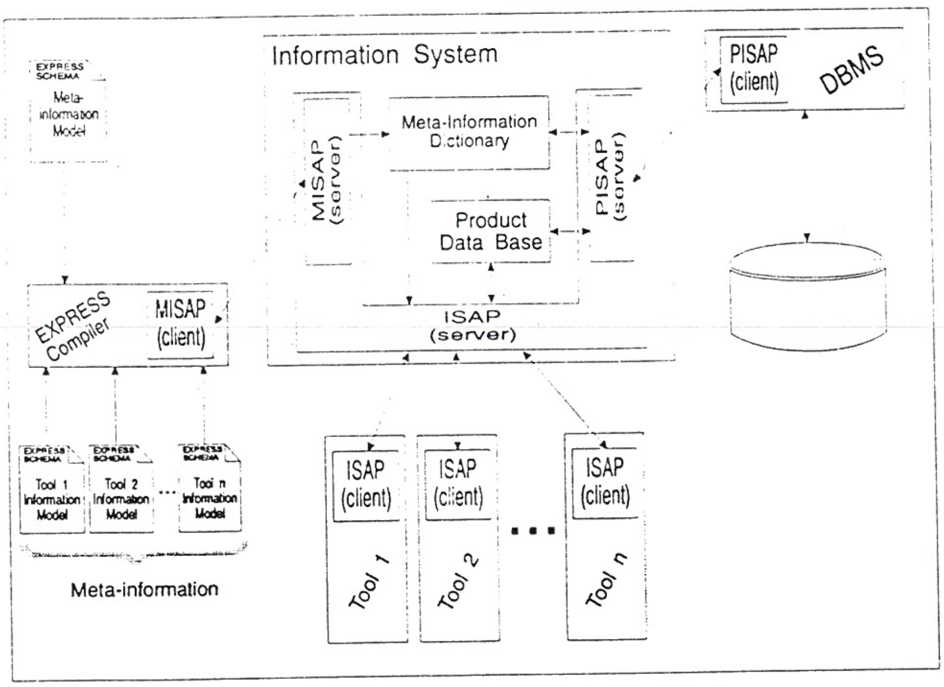
Figure depicts EXPRESS schemata containing the meta-information model and the meta-information itself. The model corresponding to the first level is stored in the meta-information dictionary, at the initialization of the IS, and is denominated bootstrap schema.

Relating to the communication protocol, two distinct parts can be identified:

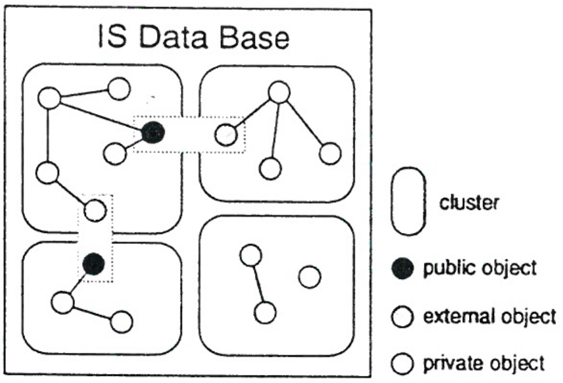
- **Client:** Library of functions allowing the communication between the server part of the protocol and the external systems. It is linked to their code and allows a virtual view of the IS operation.
- **Server:** Library of functions allowing the communication between the **client** part of the protocol and the IS, staying linked to its code, and being totally transparent to the external systems.

As already mentioned, the IS offers the local and remote capability of communication. In the local approach, the IS is directly linked to external systems, what means that the client and the server parts are linked to the executable code. We have here the IS running together with each tool. In the remote approach, the RPC (Remote Procedure Calls) service of TCP/IP (*Transmission Control Protocol / Internet Protocol*) is used. **Client** and **server** are remote processes, to run on different computers. We have, in this case, the IS running separately, physically speaking, from tools.

- CIMTOFI project's Integration Platform



Cluster of Objects



Clusters

To understand how the information is manipulated, and transmitted by the protocol of communication, the concept of clusters of objects is introduced. A cluster of objects is viewed as **a group of instances of entities (objects), belonging to the same schema, that can be manipulated as a whole**. Figure 4 presents an example of a cluster. A cluster has two different sections: HEADER and DATA (the latter one contains the objects themselves).

A cluster supports the following kinds of objects, as depicted in Figure:

- **Privates:** Objects that are not recognized in the scope of this cluster.
- **Publics**: Objects recognized in all clusters.
- **Externals**: Public objects, owned by other clusters, recognized by this cluster.

The clusters are stored by the IS in its Data Base , in the neutral format [5], and are manipulated by the **server** part of the communication protocol.

Communication Protocol Primitives

Following, a brief description of the communication protocol primitives group is presented :

Generic Functions

- Primitives for description of occurred errors

- Primitives for connection to the IS

- Transaction manipulation primitives

- Header section

- Cluster level primitives

- Object level primitives

Neutral format data manipulation

- Clusters level primitives

- Object manipulation primitives

Internal representation data manipulation

- Object level primitives

5. Integration of CAD/CAM

This section presents a STEP based interface, that uses ISAP to allow communication between a general purpose CAD system and the Information System described in the previous sections. Such an IS is part of the integration being developed by the Intelligent Robotics Center of UNINOVA - Lisbon, supported by the BRITE/EURAM project CIMTOFI - CIM to Furniture Industry.

To model information produced and consumed by the CAD system, the ISO10303 - Part 203 standard [6] was adopted. That standard contains a taxonomy of geometric and topological entities.

The interface, developed using the object oriented language C++, is a new CATIA function, that allows an interactive transfer of graphical entities .

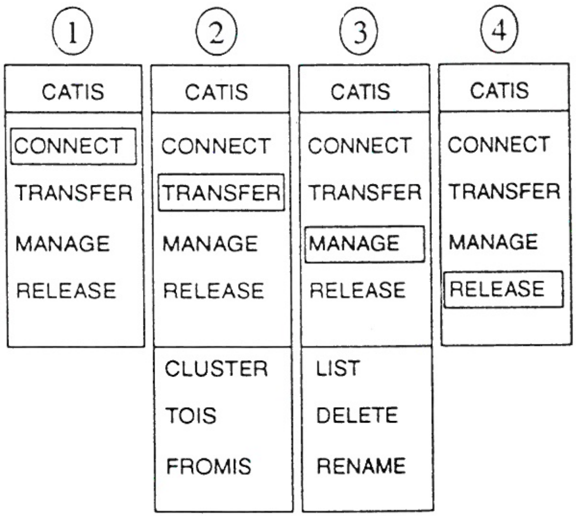
Figure 8 shows the menu hierarchy that controls the communication between CAD and Information Systems. Each item of this menu represents one task, and encapsulates communication protocol functions.

Following, the functionality of each menu item is described:

CONNECT:

Provides, by means of the communication protocol, the connection between the CAD and the Information Systems. To establish this connection, the meta-information schema to be used must be specified by the CAD interface.

Interactive Interface CATIS - menus



TRANSFER + CLUSTER:

Allows the specification of the cluster to be used in the data exchange process. If no cluster exists, a new one is created.

TRANSFER + TOIS:

Through this item the user can indicate the set of geometrical elements to be transferred to the IS. The elements are encoded according to the schema adopted. At last the encoded elements are transferred by the protocol and stored in the specified cluster.

TRANSFER + FROMIS:

Transfers the set of geometrical elements contained in the specified cluster. The incoming data are decoded and the corresponding geometrical elements are created in the current CATIA model.

MANAGE + LIST:

Presents a list of available clusters related to the adopted schema (Part 203) in the Information System.

MANAGE + DELETE:

Allows to delete a cluster from the Information.

MANAGE + RENAME:

Allows to rename a cluster.

RELEASE:

Terminates the connection between the CAD and the Information Systems.

Encoding and Decoding

The taxonomy of geometrical elements (defined by the adopted standard) was implemented using C++ classes. These classes have neutral format encoding methods, to manipulate data to be transferred to the Information System. Also, they have methods to decode the data supplied by the Information System and to create equivalent geometrical CAD system elements.

At this moment, the interface does not cover all the geometrical elements supported by the CAD system, because its purpose is just to demonstrate the described CIMTOFI integration platform. An improvement of this interface, by including geometrical elements not covered yet, will not be a too difficult task, due to the object oriented paradigm used.

6. Furniture Design/Evaluation

An EXPRESS schema used by the Furniture Design/Evaluation application as the common model of the furniture domain was defined.

The Computer Aided Furniture Design Module, baptized **FURNIT,** is a new function created under the CAE/CAD/CAM system platform. The tasks were programmed in C++. FURNIT generates furniture pieces according to the furniture model presented in the last section.

FURNIT works through menus and dialogues. Menu 2 is available through the item CREATE of menu 1. Menu 3 is available through the item STEP of menu 1. Finally, the menu 4 is available through the item TRANSFER. Following, the functionalities of FURNIT are described.

CREATE

Starts a dialogue driving the user to entering the furniture parameters. The dialogue depends on the item chosen in the menu 2. Figure 6 shows the entire sequence of steps to project a bench. For other kind of furniture the process runs similarly.

CREATE + BENCH

Creates a bench.

CREATE + CHAIR

Creates a chair.

CREATE + TABLE

Creates a table.

CREATE + WALLSHELF

Creates a wall_shelf.

CREATE + BOOKCASE

DELETE

ANALYSE

Lists the parameters of a furniture piece. Example output:

BENCH

SEAT:

WIDTH: 400.000000

DEPTH: 400.000000

HEIGHT: 30.000000

LEGS:

HEIGHT: 200.000000

RADIUS: 30.000000

POSITION: 10.000000

STEP

This is the interface to the IS. It allows the user to connect, transfer information ,and disconnect from the IS.

STEP + CONNECT

Connects to the IS using the meta-information defined in the schema simple_furniture.

STEP + TRANSFER

Allows the transfer of encoded information to the IS.

STEP + TRANSFER + CLUSTER

Defines the cluster where the IS will store neutral format encoded information.

STEP + TRANSFER + TRANSFER

Scans the internal representation of the chosen piece of furniture, generates the neutral format encoding [1], and transfers it to the specified cluster.

```
STEP;
HEADER;
FILE_NAME('/u/catia/gii/FURNITURE/clusters/banco',
'Fri May 21 12:23:17 1993',
    ('catia'),
('UNINOVA'),
'STEP VER 1',
'FURNIT VER 0',
' CATIA VERSION 3 RELEASE 2.3',
        'to_verification');
FILE_DESCRIPTION(('This Cluster contains a BENCH description',
'M0000002',
'FURNITURE FUNCTION WORK MODEL'),'1');
FILE_SCHEMA('SIMPLE_FURNITURE');
ENDSEC;
DATA;
#1=DIMENSIONAL_EXPONENTS(1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0);
#2=LENGTH_UNIT(.MILLIMETER.,#1);
#3=&SCOPE
    #4=&SCOPE

#5=LENGTH_MEASURE(400.0,#2);

#6=LENGTH_MEASURE(400.0,#2);

#7=LENGTH_MEASURE(30.0,#2);
    ENDSCOPE SLAB(#5,#6,#7);
#8=&SCOPE

#9=LENGTH_MEASURE(200.0,#2);
```

#10=LENGTH_MEASURE(30.0,#2);

ENDSCOPE LEG(#9,#10);

ENDSCOPE BENCH('BENCH XPTO',10.0,#4,#8);

ENDSEC;
ENDSTEP;
STEP + DISCONN

Disconnects from the IS.

Following the same model a rule based expert system for the furniture design evaluation, **FEExpSys**, was developed.

The system's user interface was implemented under the XVT[1] application program interface software. It made it possible to run **FEExpSys** on a Motif XWindows Workstation.

FEExpSys has two major parts: An ISAP based interface to the IS and a Nexpert[2] based knowledge processor. Next Figures depict FEExpSys' major windows.

This implementation of FEExpSys contemplates the communication in the IS - FEExpSys direction. In other words, FEExpSys does not send information to the IS. It only receives information from the IS.

The interface is accessible by means of the Cluster submenu of the major window's menubar. Three options are available in this submenu: **Connect I.S.**, **Load Cluster** and **Disconnect I.S.**.

Connect I.S.

Connects to the IS using the meta-information defined in the schema *simple furniture*.

Load Cluster

Reads a cluster containing the neutral format representation of a furniture piece and creates an internal representation. This function opens a dialog box and lists all available clusters stored in the information system.

Disconnect I.S.

Disconnects from the IS.

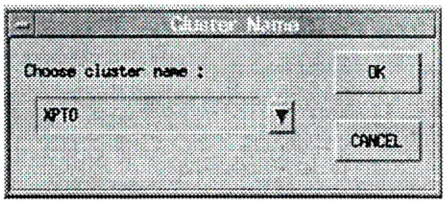
---

[1] XVT is a product of XVT Software Inc.
[2] Nexpert is a product of Neuron Data Inc.

The FEExpSys' Knowledge Processor is based on the Nexpert inference engine. This feature can be accessed through the item **Knowledge** of the major window's menubar.

The FEExpSys' data structure was automatically generated through GENKB, an utility available in the integration platform. This operation was carried out by scanning the *simple_furniture* meta-information and generating a corresponding Nexpert furniture domain representation. For more information about

Load Cluster Dialog Box



GENKB, see [3].

The furniture domain representation is based on classes, objects and slots. Objects are instances of Classes, which have properties, called slots. Slots are used to store all the information FEExpSys gathers from the IS.

Nexpert Rule's Structure

Left Hand Side	Hypothesis
	Right Hand Side
Conditions	Actions

FEExpSys supports a set of rules which contains the furniture domain knowledge. Rule's structure is shown in Figure. The conditions represent a series of tests to determine whether the hypothesis is true or not. If all conditions hold, then the hypothesis is set to true and all actions are executed.

These rules manipulate the slots, and through pattern matching and interpretations carry out the hypothesis confirmations. In the example illustrated in Figure 11, the rule has three conditions. The first one, through pattern matching, looks for an instance of the class bench and, if any, assigns the content of the instance's slot legs to the variable legobj. This variable stores, in this case, the name of an instance of the class leg.

The second condition, by means of interpretation, assigns the content of the slot height of the object whose name is stored in the variable legobj, to the variable heightobj.

Finally, the third condition, by means of interpretation, tests if the slot amount of the object whose name is stored in the variable heightobj, contains a value less than 210 millimeters.

Provided that the conditions are true the hypothesis is confirmed and the system executes the action "applyMessage", that sends the message "rule_0001 (Ergonomics) - The bench leg seems to be too low" to the user interface.

Figure 10 illustrates the Dialog Box used to apply the knowledge on the information received from the IS.

7. Current Work- Further Development

In this version, the Information System is based on the ONTOS[3] OODBMS (Object Oriented Data Base Management System). Meta-Information Dictionary, and Product Data Base are physically implemented on this DBMS. In the future, the Information System will be independent of any external storage system (a simple and dedicated DBMS will be developed).

The EXPRESS compiler generates meta-information to be directly inserted into the database. The next step will be to model the express compiler in EXPRESS, and then use the neutral format to store the Meta Information. So, it will be possible, using only the ISAP primitives, to manipulate the information and meta-information on neutral format.

As a result of the development of the CAD systems interface, it is intended to cover the geometrical elements modelled by the Part 203 and implement interfaces similar to other CAD systems - now we have developed Information System interfaces to IBM and Intergraph CAD systems, using the same interface philosophy.

Beside the development of CATIS and I-EMSIS, a Knowledge Based System to reason on furniture design and manufacturing, is being implemented. This tool is integrated through the described platform, and consumes information produced by a CAD system furniture design function.

A generator of automatic data structures, and basic object manipulation primitives will be developed. This tool will produce code for ONTOS, C++, C and SQL platforms, from the EXPRESS specification.

8. Conclusions

This paper discusses tool integration into CIM environments, in order to contribute to adopting international standards. ISO 10303 - STEP was studied and it was considered the largest initiative concerning the entire product life modelling and data communication.

An integration platform, based on this standard, developed on the BRITE/EURAM CIMTOFI project, is described.

It is the interfaces, developed for integrating applications using the platform, which make one functionality , as the CAD system integration described. Modification of data access methods being used by the tools, is not needed.

However, we hope that, in near future, the adoption of international standards will make the tools incorporate this functionality as the main data access method, thus eliminating the "Babel Tower effect", mentioned previously.

In the counterflow of this tendency , facts are identified such as the models' complexity needed to cover the whole product's life cycle, and the slowness that several involved areas are being contemplated with the corresponding standard models.

Modelling tools (EXPRESS/EXPRESS-G) and methods to support data in a standard format (Neutral File) are properly developed. So, large investment from enterprises and research centres is imperative in order to develop these standard models and tools and to manipulate them in an easy way. Otherwise, the so far investments will be lost, as well the STEP credibility.

---

[3] ONTOS OODBMS is an Ontos Inc. product

Due to the high complexity of CIM question, it seems that the most acceptable strategy is "to divide and conquer". Individual initiatives, by the giants of the information industry, have not produced acceptable solutions, globally. In our opinion, enterprises specialized in software to manufacturing environment must make strong efforts for the development of information models on their application areas (CAD, CAM, CAPP, CAP, CAE, all kinds of management tools, etc.). This is possible through the modelling tools released by the standard STEP. Such enterprises would develop their applications based in the same standard, enabling an easy and total integration of tools.

REFERENCES

1. BARRETO, M. P. , **CIM: um alvo em constante movimento**, Anais da Reuniao do programa CYTED-D, , Caracas, Venezuela, September 1991.

2. CAMARINHA- MATOS, L.M. and OSORIO, L., **CIM Information Management System: An Express-Based Integration Platform**, IFAC - Workshop on CIM in Process and Manufacturing Industry, Espoo, Finland, 23-25 November 1992.

3. ISO TC184/SC4/WG7 N262, ISO 10303 - Part1 - **Overview and Fundamental Principles**, 15 September 1992.

4. ISO TC184/SC4/WG5, ISO 10303 - Part11 - **EXPRESS Language Reference Manual**, 29 April 1991.

5. ISO TC184/SC4/WG7 N262, ISO 10303 - Part21 - **Clear Text Encoding of the Exchange Structure**, 15 October 1992.

6. ISO TC184/SC4/WG7 N262, ISO 10303 - Part22 - **Standard Data Interface Specification**, 15 October 1992.

7. ISO TC184/SC4/WG4 , ISO 10303 - Part203 - Configuration Controller Design, 10 September 1992.

8. WESTON, R.H., with collaboration of Aprotec International, Loughborough University of Technology and Wire Systems, **A Requirements Definition for Software Interoperability in Multi-Vendor Manufacturing Systems,** 30 September 1992.