# *Interoperability between CAD and simulation models,*

O. Hamri*,**, J. C. Leon*,
F. Giannini**, B. Falcidieno**

*Laboratoire Sols Solides Structures*
*Projet Conception Intégrée — CNRS/INPG/UJF*
*Domaine Universitaire, BP 53, 38041 Grenoble Cedex 9*
*{jean-philippe.pernot, stephane.guillet, jean-claude.leon}@hmg.inpg.fr*

**Istituto di Matematica Applicata e Tecnologie Informatiche — CNR*
*Department of Genova*
*6, via de Marini, 16149 Genova, Italia*
*{jean, falcidieno, giannini}@ge.imati.cnr.it*

**Technical Report CNR IMATI-Ge n° 1/04**

# **Research report:** Interoperability between CAD and simulation models.

09/03/2004

The work has been carried out within a bi-lateral collaboration agreement between L3S of the INPG in Grenoble (France) and CNR IMATI-GE in Genova (ITALY).

## Contents

# 1. Introduction

One of the main problems found in the passage from Computer Aided Design (CAD) to Computer Aided Engineering (CAE) is the lack of intersecting application space between the two groups. CAD models are typically generated to create the product shape without prior knowledge of their effects on downstream CAE applications like Finite Element (FE) mesh generation. To generate a FE model, the CAD geometry has to be adapted to suits the hypotheses of the needed mechanical model. This task cannot be performed solely on the basis of a geometric data [1-5], but require also engineering expertise [6] to supply the necessary additional information, such as Boundary Conditions (BCs). This preparation process is designated here as an *a priori* model adaptation since it is strongly based on the user's know-how to specify the FE simulation requirements. Therefore, a direct automatic transition from a CAD model to a finite element model is not feasible [7, 8].

Generally, the difficulty in the analysis model preparation is the generation of a mesh for complex or detailed models that constitutes the basis for simulations. The mesh may also become too complex or difficult to adapt for more different simulations, using different categories of meshes (e.g. volume or surface) or hypotheses (shell, plate, beams, triangles, quadrangles, ...). Moreover, no software or efficient data-processing approaches currently exist that make possible to adapt a pre-existing mesh or polyhedral model of a component to speed up the overall simulation process [9]. The waste of time and the cost caused by such a situation impact on the design cycle when trying to reduce its duration or to parallelize some tasks. Feature approaches have been extensively developed over the past years [10, 11] to insert parameterised semantic elements characterising the product, unfortunately mainly restricted to CAD/CAM integration. From the CAE point of view, few works have addressed the problem of maintaining the information between a FE model and the CAD model representing the shape of a simulation model [12, 13] by mainly using attribute structures. In our work, the previous concepts of feature and attribute are combined to originate the concept of simplification features to address the shape modification requirements needed during the FE model preparation phase.

Since in our application scenario we aim to deal with various types of input data, such as CAD models and scanned objects, the polyhedral model seems to be the best starting point for the preparation phase of FE meshes, as opposed to the use of a CAD model [6]. Anyhow, in order to take advantages of all the available information, we propose a common topology approach which allow the management of the various possible input models, especially form feature models, to increase the efficiency of an analysis model preparation process for structural analysis.

# 2. Analysis model preparation (the first appraoch):

## *2.1. From CAD model*

Currently, FE meshes and analysis models are generated from CAD models during the product design phase (see figure 2). Such a preparation phase often requires the use of geometric treatments for adapting or idealizing the model to fit the hypotheses and requirements of the FE analysis foreseen. Either based on CAD models preparation, or on FE mesh modifications, model adaptation and idealization are necessary to fit the FE analysis requirements. Recently, new approaches have been proposed to generate meshes from models prepared for rapid prototyping manufacture, namely STL files, to be able to handle new sources of data.

Starting of the CAD model (figure 1) created by the design office, this can be presented in several forms, according to objectives of analysis, for example B-Rep model in structural analysis. This model depicts the design results available at a time “t0”.

At this time, mechanical hypothesis, simulation objectives are inserted by the user to help generate the domain of study compatible with the simulation requirement; As a result the model M called case of study is obtained and enriched with boundary conditions (figure 2).

Engineer design receives this geometry in order to generate the analysis model.

The generation of this, last passes by the creation of a simulation model by a simplification and idealisation, the geometry resulting of this stage will be useful to us a basis for the mesh generation stage.

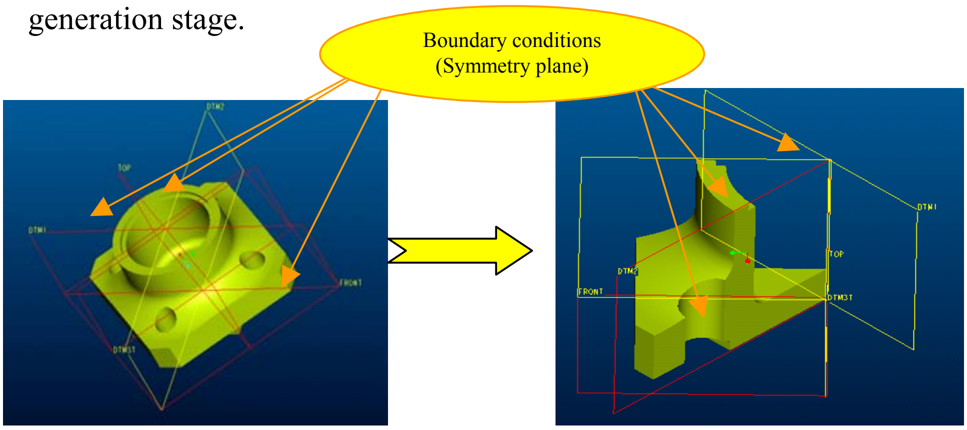



**Figure 1** Case study generation

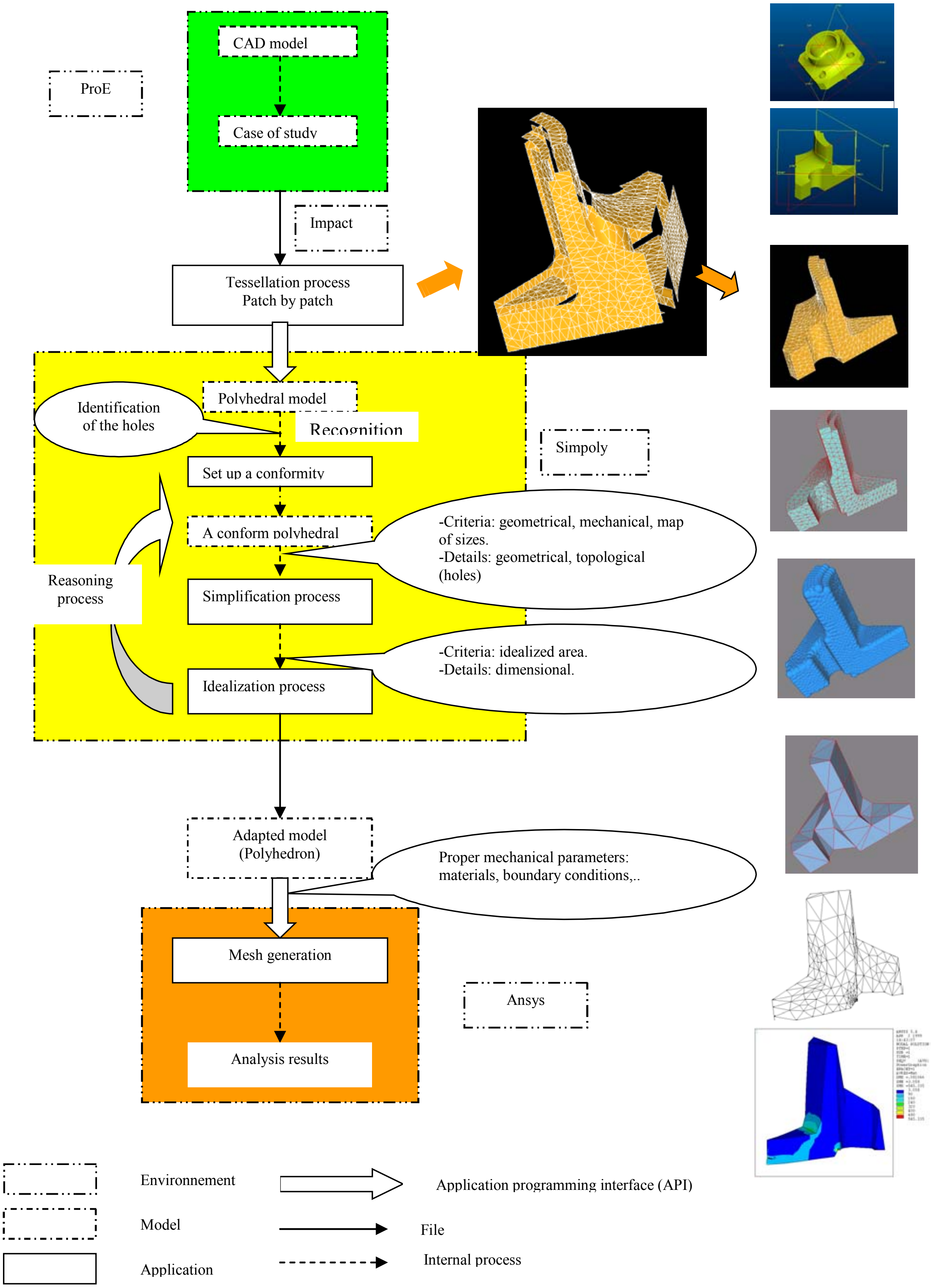



**Figure 2** Analysis model preparation

## *2.2. Tessellation process:*

In order to generate the simulation model (adapted model) the last step is to generate the intermediate model defined by the polyhedral model and obtained by the tessellation process. This process is performed in the tessellation environment which is done.
The first step of this process is imported the CAD model via STEP format (AP214 or AP203), (figure 3)
After this process the tessellation process can be performed by the definition parameters of this process (deflection, target edge length).These parameters are given by the user in order to check the density of the polyhedral model and his quality.
The result of this process is a non conform polyhedral (figure 4), because this one is performed on a patch by patch basis, then there is not exist a common parametric space has all surfaces, and also this problem is due to the data exchange because the CAD model is generated in the other environment ( CAD software, figure 1).

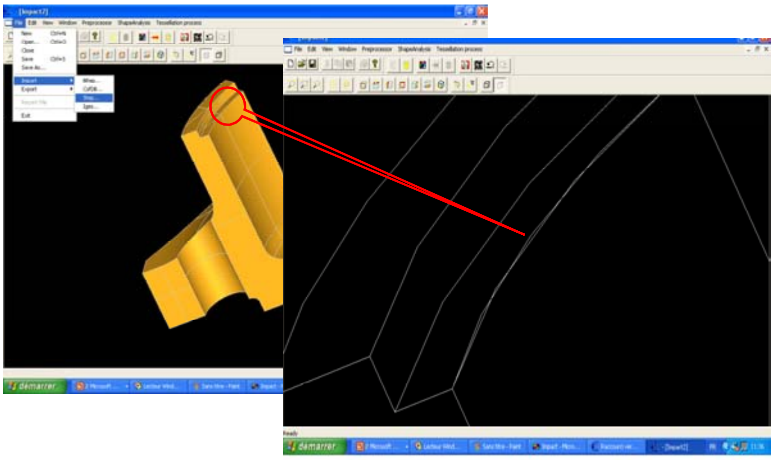


**Figure 3** Data exchange problem

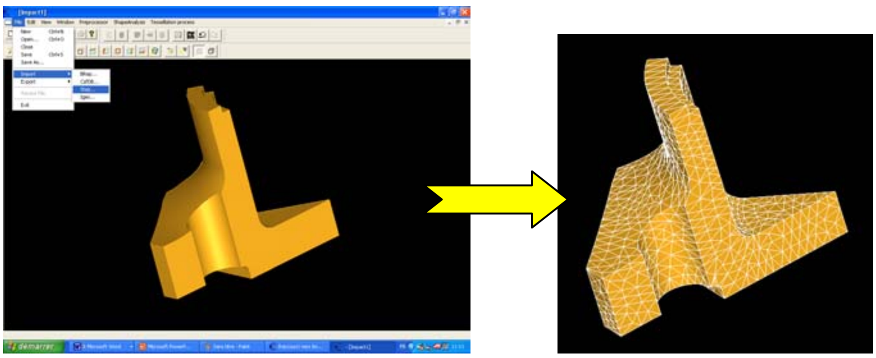


**Figure 4** Tessellate the imported CAD model

### 2.3. Set up conformity

In order to generate a conform polyhedron model; the last process is the setting up conformity. This process is performed by applying the set up operators for conformity.
This process is performed automatically without the user in Simpoly software.
For example of these operators: figure 5, and figure 6, 7.

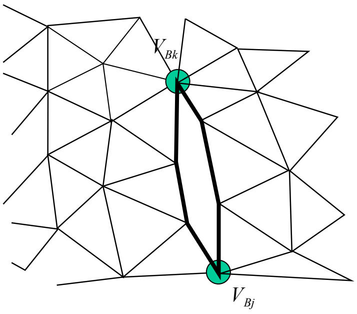



**Figure 5** Effect of the removal of coincident vertices coinciding with vertices $V_{Bk}$ and $V_{Bj}$. The connected partitions of the triangulation create elongated holes that will be treated by subsequent operators.

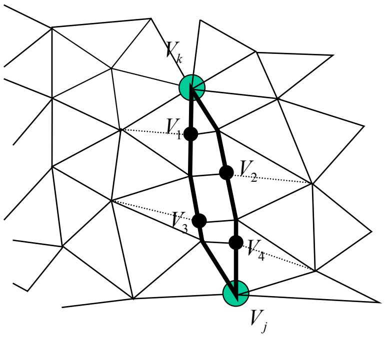
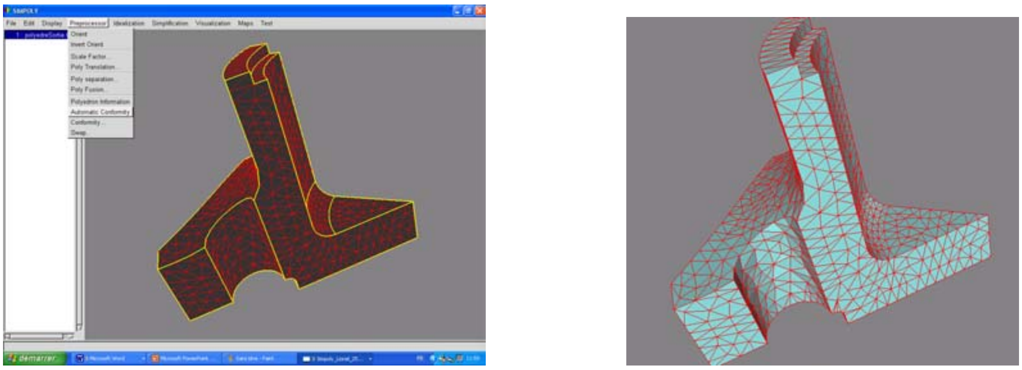



**Figure 6** An elongated loop of edges with two extreme vertices $V_k$ and $V_j$. New vertices $V_1$, $V_2$, $V_3$, $V_4$ are created as well as the associated edges. New faces are created to remove the closed loop.



a) b)

**Figure 7** Set up conformity process: a) a non conform polyhedron, b) a conform polyhedron

### 2.4. Simplification process

The last step is to generate the adapted model by simplification process in order to remove some details which are not relevant for the analysis foreseen, this process is based an iterative removal vertex and remeshing operators. Figure 8.

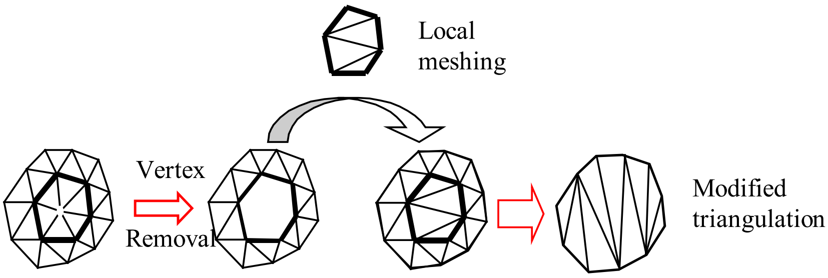



**Figure 8** Simplification process

The details are defined relatively to mechanical behavior of the structure within the context of the desired analysis.
The user defined the map of sizes in order to generate the adapted model.
This map of sizes can be understood as a discrete envelope (see figure 9) set up the initial polyhedron where the adapted polyhedron must be lie.

The simplification process may be performed with equilaterally remeshing (see figure 10 a) or by the curvature criterion. (see figure 10 b)

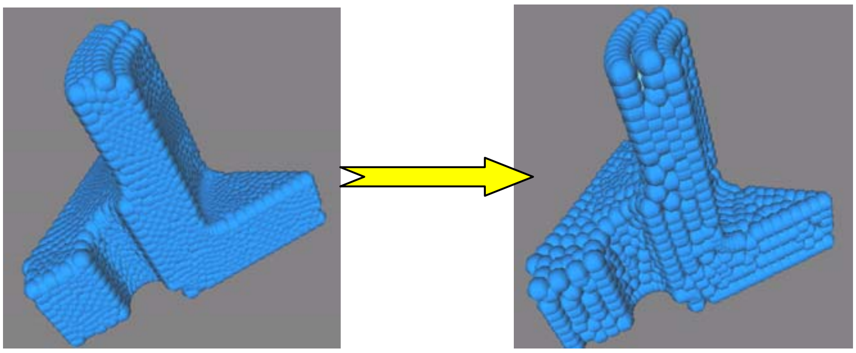
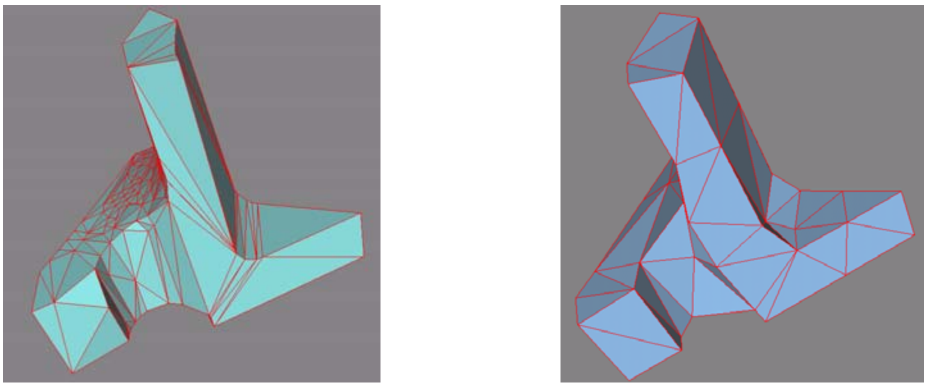


**Figure 9** specification of map of sizes by the user



a) b)

**Figure 10** Simplified models: a) curvatures criterion , b) equilateral re-meshing

The last stage is to generate the mesh of the adapted model, the boundary conditions (B.C) is transferred on the adapted polyhedron (figure 11) in order to applied on the mesh (see figure 16).
Finally the analysis results (figure 12) are obtained from the mesh of the adapted polyhedron enriched with the proper mechanical parameters.

**Figure 11** Transfer of BCs on the adapted model

At the moment the mesh process is not done.

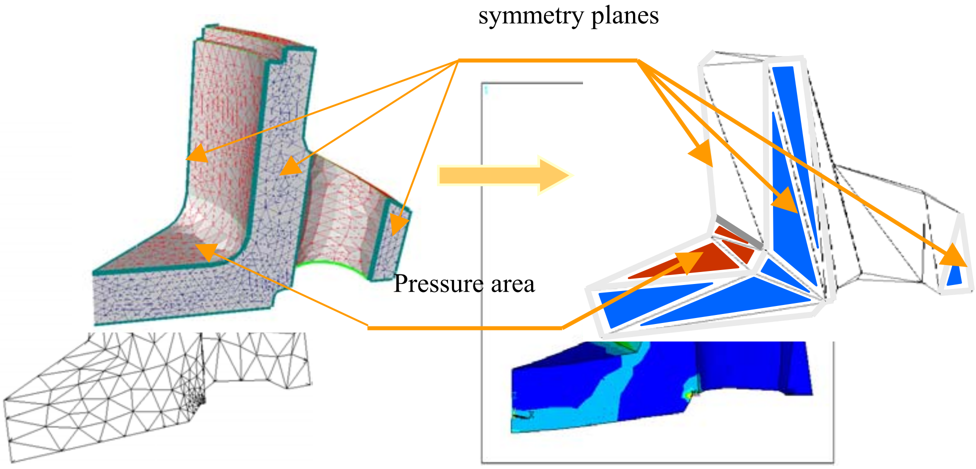



**Figure 12** Mesh generation and the analysis results

## 2. Interoperability between CAD and simulation models:

Starting from the basis of a design model at a given stage of the design process, such a model is considered as a B-Rep surface representation, i.e. a CAD model as it can be modelled with current industrial modelers (see Fig. 13). This model depicts the design results available at a time $t_0$ (see Fig. 14). At this time, mechanical hypotheses, simulation objectives, are inserted to help generate the domain of study compatible with the simulation requirements. As a result, the model $M$, called case of study (see Fig. 13), is obtained and enriched with boundary conditions. From $M$, a finite element mesh $M'$ is derived to form the basis of the structural analysis (this process is obtained while following the stages described previously see schema1). The time $(t_0 + t_1)$ is reflecting the set up of the analysis process. Finally, at this time the analysis results $A_M$ are obtained from the mesh $M'$ enriched with the proper mechanical parameters. Such a workflow illustrates the standard operations required to perform an analysis. The results $A_M$ it is necessary for validate of this process. The set up of a new analysis at $t_2$ can occur only if the process is validated ($t_2 > (t_0 + t_1)$).

Now, considering that a modification of the initial CAD model has taken place to fit new requirements or to derive a new version of the component, a new version of the case of study through the current processes of geometry update. This modification produces a model, $M''$, at. $t_2$ Here, the question is raised whether or not the mesh $M'$, derived from the model $M''$, is still acceptable to model the behaviour of the structure or if a new mesh $M'''$ needs to be produced to perform a new analysis and achieve the coherence between $M''$ and $M'''$. Hence, the interoperability between CAD models and simulation ones is achieved between models $M, M', M'', M'''$ if the coherence among them is preserved during the design process, i.e. $M'$ is attached to $M''$ or $M'''$ is attached to $M''$ depending on the analysis objectives. Such a configuration of interoperability is called one way interoperability. The interoperability between simulation and CAD models, i.e. the reverse configuration, is not part of the present work.

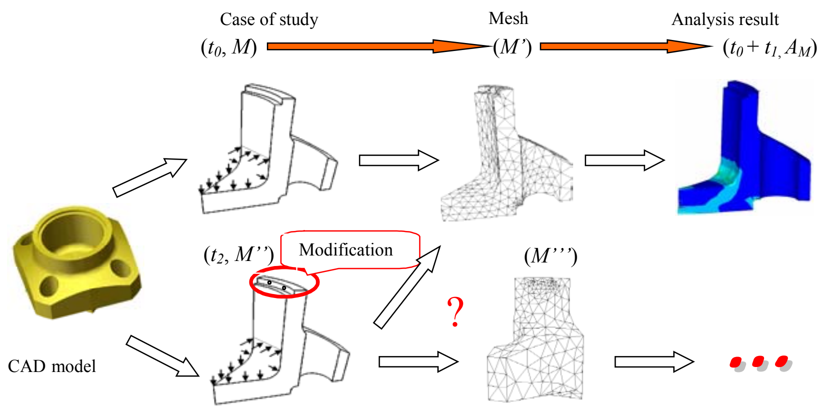



**Figure 13** Interoperability problem

## 3. The concept of the intermediate model

The intermediate model is dedicated to (see figure14):

- Ensure the link between a CAD model and a simulation model,
-
-
-
-
- Be an adequate model for mesh generation,
- Support local geometry transformations for shape adaptation.
-

### Why choose the polyhedral model us an intermediate model?

Because this model is:

- 

The polyhedral model is choose us intermediate model, because this model i :

- A simple geometric model in the field of knowledge of the analyst,
- A geometric model suited to ensure the CAD/CAE model link,
- A model which supports the local geometric transformations,
- A model usable to generate meshes
- 
- 
- 

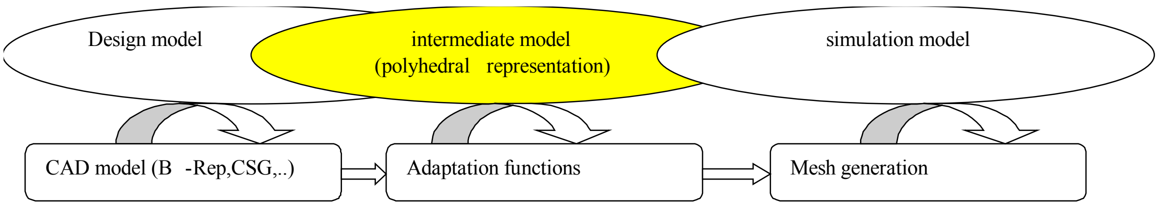



**Figure 14** The concept of intermediate model in analysis model generation context

## 4. Proposed approach

In the objective to ensure a link between the CAD and the simulation models, our approach consists has to exploit a history of construction of CAD model based mainly on a “features “ approach because this approach has the advantages according to (schema2)

- To reduce the complexity of the simplification task by the use of the form features as high level entities (having encapsulated geometrical properties). Indeed this task becomes complicated when it is applied starting from a solid model (B-Rep) and requires tiresome geometrical calculations for the manipulating of the entities of low levels such as the vertex, edges and the faces, therefore we can use this approach upstream of simplification process in order to realised certain adaptation directly to the level of CAD model (suppression of the holes, modifications of connection between the geometrical entities, ..), and in the case where known information is only of low level, it is has to say, if it are not sufficient to characterise features directly existing, then can nevertheless to complete the analyses to take from the polyhedral representation to identify the three classes of details (geometrical, topological, dimensional), this information can be has character geometrical (taking into account of the decomposition of a square of a geometry B-Rep, existence of particular surface, ..) or technological (information on the existing parameters features of the model,...).

- To avoid the loss of information between the CAD model and its simplified model, this relates to the form features considered to be non relevant and which consequently do not appear in the simplified model. That therefore are not lost, but remain in a form abstracted in the initial model and can be reintroduced has any moment. This remark is very significant because the keeping of information in the simplification process guarantees the possibility of found the initial geometry of the model.

### *4.1. Analysis model preparation through Feature approach*

CAD and FEA are two significantly different disciplines, and hence they demand considerably different object model representations. As a result, models generated by CAD systems are often unsuitable for analysis needs, requiring multiple editing and shape adjustments before the creation of the foreseen FE mesh and the insertion of the required Boundary Conditions (BCs). Among the problems requiring editing, one can enumerate:

- Incomplete and inconsistent topology and geometry descriptions (due to data exchange between different software systems). This may be occurring because of differences in surface definitions between CAD packages. For instance, while Pro/Engineer defines its surfaces with twelve different surfaces types, ACIS uses only five ,
- Irrelevant details which severely complicate the meshing phase while having no significant influence over the mechanical simulation performed,
- Idealisation of some areas of the model to fit with the objectives of the simulation, e.g. idealise a volume area into a surface one to get compatibility with a mechanical shell behaviour ,
- Insertion of BCs in order to generate the 'case of study' satisfying the mechanical hypotheses set for the simulation model, e.g. pressure areas applied to the studied component.

For the previous reasons, an automatic conversion of the CAD model into an analysis one is not possible. Most of the time, such a conversion requires at first, the selection of a sub-domain of the object on which the analysis can provide results comparable with those obtained on the whole object, this is normally indicated as the case of study definition.

To avoid the editing of complex CAD models and to ease the integration of the preparation process into a wide variety of design configurations and input data, the approach here proposed is based on a polyhedral model. Such a type of model is compatible with the requirements for simulation models: simulation models are based on FE meshes, which are discretized models similar to polyhedral models. This approach has been already validated in the preparation of complex FE model for the thermal analysis of the A380 Airbus aircraft cockpit. Here, the use of the polyhedral model during the preparation phase has reduced by more than three times the amount of time required for this phase compared to the current practice based on CAD model editing prior to the FE mesh generation phase [15].

Figure 15 illustrates the structure of the proposed model preparation process, starting from a CAD model (possibly feature-based) enriched with additional mechanical information (A) and finishing with standard FE mesh generators and solvers (C). This structure clearly shows that the preparation process can be inserted in any CAD-FEA software environment. Standard CAD modellers are distinguished from feature-based ones to clearly cover all the possible industrial configurations, i.e. CAD modellers that can use a variable range of form feature primitives. Here, we restricted to B-Rep models since always available through standard data exchange formats (e.g. IGES or STEP). The preparation process (B) can be applied to different input polyhedrons, i.e. evaluated from CAD models, digitized models, pre-existing FE meshes. Even if this capability is not under focus here, it is shown to reveal the overall model preparation scheme to demonstrate how this scheme behaves depending on the amount of information existing in the input model. Illustrations of examples based on digitized models or pre-existing FE meshes can be found in.

Figure 15 illustrates the structure of the proposed model preparation process, starting from a CAD model (possibly feature-based) enriched with additional mechanical information (A) and finishing with standard FE mesh generators and solvers (C). This structure clearly shows that the preparation process can be inserted in any CAD-FEA software environment. Standard CAD modellers are distinguished from feature-based ones to clearly cover all the possible industrial configurations, i.e. CAD modellers that can use a variable range of form feature primitives. Here, we restricted to B-Rep models since always available through standard data exchange formats (e.g. IGES or STEP). The preparation process (B) can be applied to different input polyhedrons, i.e. evaluated from CAD models, digitized models, pre-existing FE meshes. Even if this capability is not under focus here, it is shown to reveal the overall model preparation scheme to demonstrate how this scheme behaves depending on the amount of information existing in the input model. Illustrations of examples based on digitized models or pre-existing FE meshes can be found in.

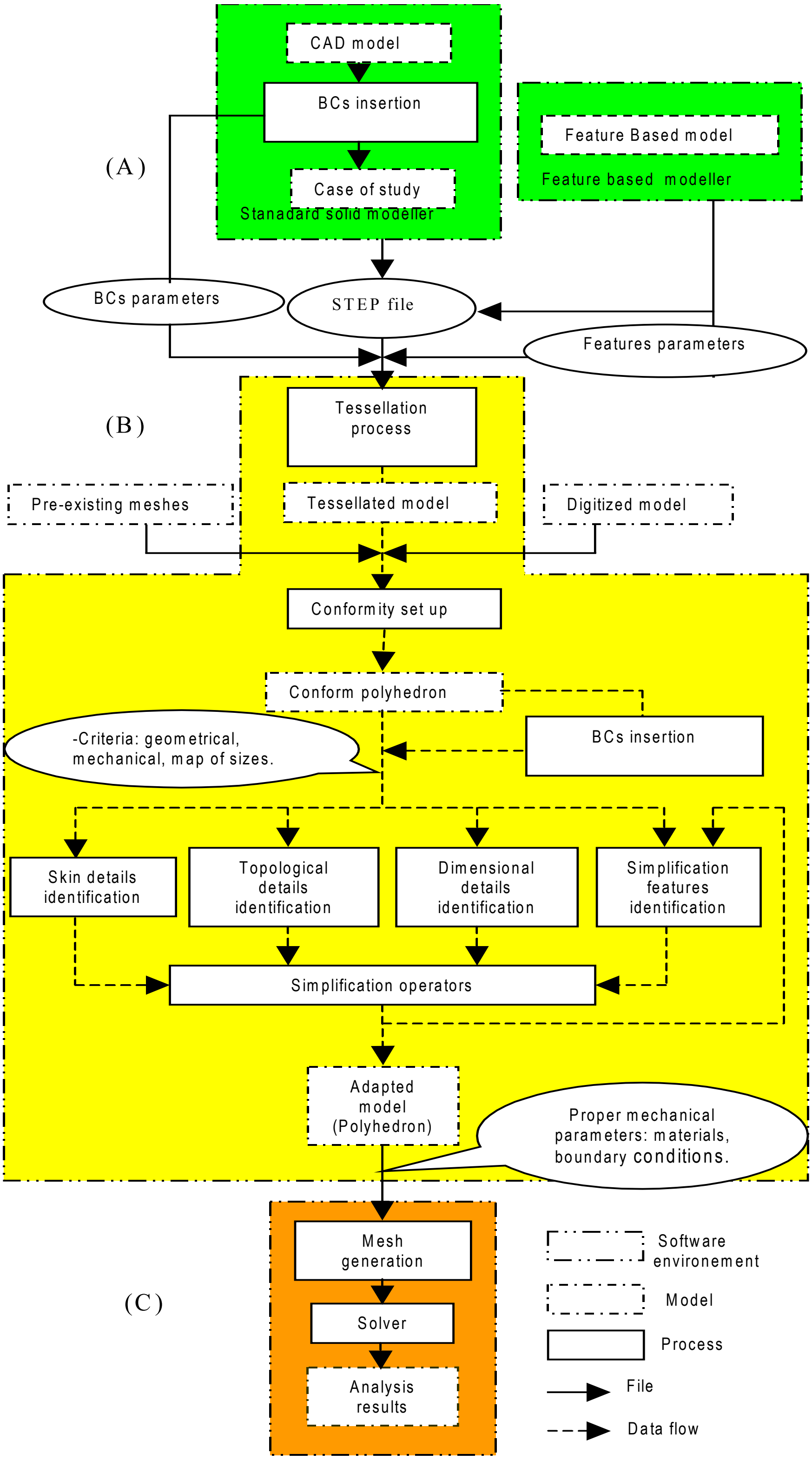



**Figure 15** Data flow for analysis model preparation

### 4.1.1. From CAD model:

At the modeller level (A), standard solid modellers are associated with a process, namely the insertion of BCs to produce the case of study. The case of study designates the considered component's sub domain that is required for the analysis in accordance with the hypotheses as well as the location of prescribed forces and/or displacements defining the loading conditions of the component. The generation of a case of study is often based on some of the BCs required to set up or to simplify the analysis model. Symmetry planes, pressure areas, force locations are the most recurrent BCs. Most of them are inserted through the use (and possibly creation) of adequate geometric elements; thus motivating at level (A) the BCs insertion process, which can be considered as a specific application module.

Geometric operators of standard CAD systems as well as specific ones are used to create the geometric model of the case of study, which is usually non-manifold. Whilst the geometric model can be exported at level (B) through a standard format such as STEP, this is not possible for the added BCs. In fact, BCs parameters, which coincide with the mechanical parameters that are attached to the BCs, i.e. pressure values, forces components, are not treated in the STEP application protocols used by most of CAD systems, i.e. AP203 and AP214. Thus BC parameters currently need to be transferred through specific file formats to input them into the model preparation environment (B).

Still at level (A), feature based modellers are distinguished from standard solid modellers because there are still limitations in handling feature information in standard formats. Similarly not all the systems treat the same set of features primitives. As a consequence three categories of data are returned from these modellers:

- A geometric model of the component that can be exchanged through a STEP file,
- A set of feature parameters to describe the form features that requires a specific file format since STEP files still cannot incorporate such type of information,
- A set of BC parameters to describe the possibly defined mechanical parameters.

### 4.1.2. The model preparation level:

At this point, a large range of possible configurations for producing CAD models of components has been addressed to characterize the input data of the simplification process taking place at level (B). The resulting data can vary from a standard B-Rep NURBS model to a similar model enriched with BC data, to a model enriched by a feature-based structure.

The model preparation environment for the above range of data starts with a tessellation process to produce the polyhedral representation required for the detail removal process. Rather than using a tessellation process integrated into CAD modellers, where the criteria used can vary widely and the control parameters may not be suited to the model preparation process, i.e. the lack of control parameters may produce very sharp triangles that are not compatible with the range and accuracy of the simplification operators applied later on. A tessellation process, independent of any CAD software, has been set up using Ruppert's algorithm adopting an edge length criterion while avoiding degenerated triangles.

Incorporating the tessellation process into the model preparation phase offers also the possibility to relate its control parameters to the detail identification process taking place later on. However, it should be mentioned that there is not yet any clear specification of operators to bind efficiently the tessellation to the simplification processes. Thus the tessellation process by itself needs to be controlled by the mechanical engineer in charge of the simulation to ensure that the discretization of the CAD model is somehow compatible with the size of the FE required in the FE mesh.

The tessellation process incorporates also the task of transferring BC attributes onto the polyhedral model generated as well as some form feature information, whenever they exist, to enhance the input data for the simplification operators or preserve the appropriate data contributing to the description of the required simulation model. In addition, the tessellation process can maintain the information of which triangle edges are derived from edges of two adjacent faces in the B-Rep... Such edges are designated as *homologous* edges. This distinguish them from boundary edges encountered in digitized models, which often result from shadow areas and do not convey the same meaning as B-Rep boundary faces for the conformity set up process as described later.

Our tessellation is performed on a patch by patch basis, hence the resulting polyhedron is not conform, i.e. the triangles belonging to the boundaries of two distinct patches are not connected and reflect the accuracy of the patch connectivity of the considered CAD modeller. Such non conformities come from the distinction between the description of the topology of an object as specified in a CAD modeller and the corresponding different geometric descriptions of two adjacent B-Rep faces. To produce a conform polyhedron as needed by the simplification process, a conformity set up process is mandatory for the tessellated CAD models as well as digitized models. Even pre-existing FE meshes may require such a treatment when the objective it to set up a model from several parts (see Fig. 16).

In the case of CAD input making a model conform means that the topology of the polyhedral model must be identical to that of the initial B-Rep model. In other words, for a two-manifold closed surface representing the boundary of a component, each edge of the polyhedron must be connected to exactly two faces.

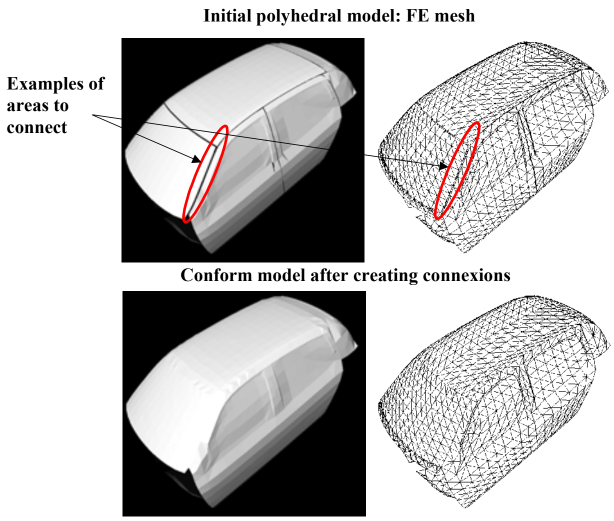



**Figure 16** Conformity set up to connect a set of components from a car body (courtesy Renault).

Setting up conformity for a tessellated model is a critical task. Several approaches have already been proposed by Barequet and al. [21, 22] to repair CAD models and concentrate on two-manifold boundaries to avoid generating non-manifold configurations. In our approach, a set of operators has been created to modify the input triangulation, which. Has proven efficient on complex models, even when no adjacency information, like the B-Rep face adjacency mentioned earlier, are available (cf. the results obtained for the A380 aircraft cockpit).

This set of operators set up [9] to handle general models (either manifold or non-manifold) can create temporarily non-manifold geometry before vanishing through the application of subsequent operators. Figure 4c, d illustrates one such operator: removal of all vertices coinciding with others within a tolerance $\varepsilon_v$ specified by the user.

This operator is uniformly applied to All the vertices of the triangulation, i.e. $\left\|V_i - V_j\right\| \le \varepsilon_v, \forall\, i, j, \quad i \ne j$ where $V_i$ designates the 3D coordinates of the i$^{th}$ vertex. Locally, this operator can generate a non-manifold triangulation. The value of $\varepsilon_V$ is critical for the efficiency of the procedure. A particular version of this operator takes into account the category of vertices used for comparing their 3D positions. In this case, only boundary vertices $V_{BK}$ are included in the treatment. Such a version of the operator is useful for creating a first connection between independent partitions of a surface (see Figure 3c, d).

Other operators used to perform this healing process are:

- Removal of degenerated edges,
- Removal of degenerated faces,
- Removal of duplicated edges,
- Removal of duplicated faces,
- Stitching gaps represented by elongated closed contours according to a user prescribed tolerance.

This set of operators does not incorporate any attribute such as *homologous edges* and *vertices*. These operators are not considered as generic and capable of handling any kind of defects but anyhow they have been evaluated on a series of industrial models produced by different CAD systems or scanning devices producing good results. As a result, they o can be considered as a sequence of treatments adequate for a given modeller, i.e. adapted to a specific set of tolerances compatible with a given modeller.

Now, taking into account the component topology described in the STEP file and the *homology* between edges and vertices, the previous set of operators can rigorously identify the *homologous vertices* to merge them, to stitch the appropriate *homologous edge* sets, therefore robustly producing a conform polyhedron having a topology identical to the component described in the STEP file. In addition, these operations can produce simultaneously the new attributes to mark *uniquely* the polyhedron edges corresponding to the topological edges defining the B-Rep model of the component in the CAD modeller. As a result, the conform polyhedron can be bound with the B-Rep topology of the CAD model and if available, with the feature and BC attributes which may be input to the preparation process. In the later case, the conform polyhedron is enriched with the CAD B-Rep topology, feature and BC data as well as attributes concerning the geometric primitives attached to the B-Rep faces in the STEP file.

If not already available, the BC parameters are now incorporated to the polyhedral model as depicted on Fig. 1. Though, the model here is a polyhedron, the operators required are similar to those described at the level (A) of Fig. 15. Adding the BCs at this stage is mandatory when the input polyhedron comes from digitization or is a pre-existing FE mesh. In addition it is necessary to specify the so-called a map of FE sizes to characterize the shape details for the analysis foreseen [6, 7]. This mechanical concept is employed to identify the geometric details since, for example, FE sizes state the distribution of strain energy in the structure for a given load configuration. Large FE sizes, i.e. large spheres, compared to the local edge lengths characterize the details that need to be removed to get a FE mesh compatible with the object shape. Such a map of FE sizes is user-defined according to the *a priori* approach described here. It represents a discrete envelope around the polyhedron which monitors the simplification operators, see fig. 17.a.

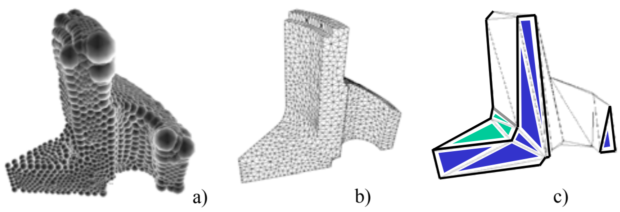



**Figure 17** Examples of map of FE sizes (a) attached to a polyhedron (b). Simplified model (c) incorporating the BCs.

The next step is the generation of the adapted model obtained by through simplification processes according to the simulation objectives.

Depending on the type of modifications that must be carried out on the polyhedral model to remove details, three classes categorize the details and their associated operators [9] as shown in Fig.18:

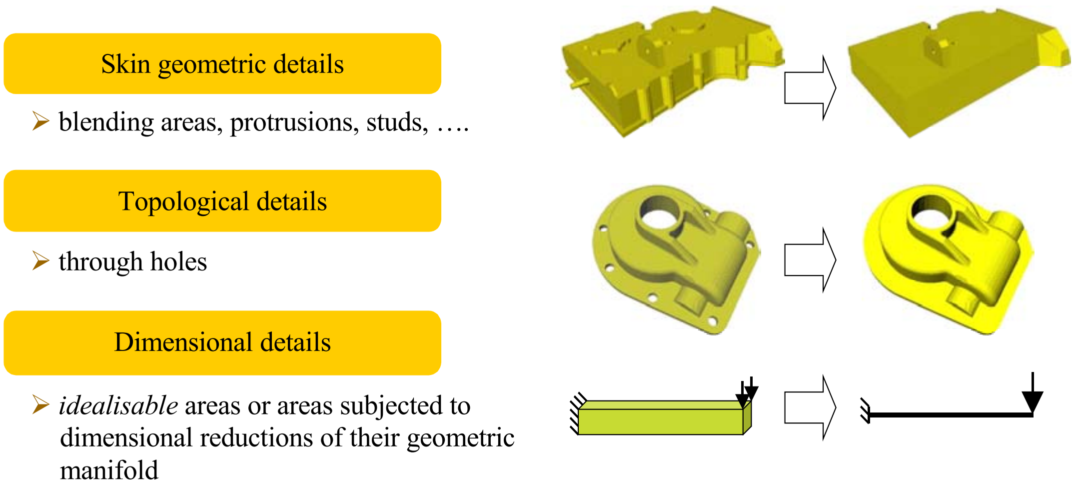



**Figure 18** Examples of the three categories of details participating to the simplification process.

1. Skin, designates those details that can be removed by performing only continuous transformations like deforming a clay model, and its associated removal operator is based on a decimation principle, i.e. an 'iterative vertex removal and local remeshing' [7, 8]. The identification criteria of this category of details, is based on the analysis of the influence of each vertex of the polyhedron on the geometric model of the object. The decimation operator is based on the 'error zone' concept. A spherical error zone is assigned to each vertex of the initial model. The radius of these spheres is defined through the user- specified FE map of sizes by mean of interpolation functions that smoothly assign radius values at each vertex using, as input, the user-specified values at key points.

The set of error zones can be understood as a discrete envelope set up around the initial polyhedron where the decimated polyhedron must lie [9]. To this end, the vertex removal process is combined with an inheritance process such that error zones attached to the removed vertices are kept active entities of the discrete envelope during the decimation process. At each iteration, the inheritance mechanism is achieved with a redistribution process of the error zones over the faces resulting from the remeshing phase.

The combination of the vertex removal operator and the inheritance process help formulate the shape restoration criterion to express that the decimated polyhedron stays always within the discrete envelope that expresses a subset of the simulation objectives.

The remeshing operator applied at each iteration creates a new geometry from the contour polygon of the candidate vertex. The shape restoration criterion is then applied to determine whether the vertex can be removed or not. If the geometry of the initial model is correctly restored, the current model is updated using the previously created mesh of the 3D contour polygon [8].

2. Topological, designates those details affecting the genus of the object, like through holes that cannot be removed by continuously deforming the object surface. The goal of the hole removal operator is to locate and remove automatically through holes from the object. These operators are taking benefit of the skin detail removal operator prior to use properties of fundamental group of curves and surfaces, and identify the set of faces forming the surface of the hole to remove [14].

The goal of the hole removal operator is to locate and remove automatically through holes from the polyhedral model describing the object. This operator is applied to through holes placed in closed two-manifold sub domains of a polyhedron, where each edge is exactly adjacent to two faces, it means that no topological criterion can be used to distinguish holes edges from others. Since geometric criteria are hardly robust and general to detect edges delimiting hole faces, the localisation phase cannot be performed on the input polyhedron. Indeed, this phase takes place during the decimation process where specific face-edge configurations are characterised from a topological point of view. Thus, the decimation process adds a dynamic insight which can be exploited to robustly locate through holes in connection with the error size value specified through the FE map of sizes. For holes corresponding to details, the skin detail simplification process reduces a hole to its basic polyhedral shape, i.e. an open polyhedron with a triangular basis.

When each hole has been identified and characterised by a connected subset of faces, the topological operator removes all the nodes, edges, and faces defining this subset and generates two new faces based on each edge loop defining each hole boundary in order to close it.

As stated, through holes are removed using a combination of topological properties, decimation process and a part of simulation objectives expressed through the FE map of sizes.

3. Abstraction refers to those areas of the object that can be idealised by using 2D or 1D geometry, e.g. lamina or polylines. Their associated removal operators reduce locally the dimension of the geometric manifold of the component, using pairing operations between geometric entities of a given component.

The idealisation process operates in two stages and refers to the abstraction of sub domains of the object. The first step may be called geometrical idealisation, since it uses geometrical criteria as the only input to proceed. It consists of a loop process ran all over the geometry. The process is as follows: first the geometry is analysed to determine a starting point for idealisation, depending on the specified geometrical criteria: this is the localisation process. It tells where idealisable areas are based on a concept of error zones that indicate whether some dimensions of FE elements highlight directions able to accept dimensional reduction. This is typically the case when a structure can be assigned shell behaviour where the thickness direction can be removed to create an open surface. Second, the idealisation algorithm itself is run over this area, creating a surface area and a locally non-manifold geometry. The automated process is reiterated over the complete geometry until it is unable to find any valid area to be idealised.

The second step requires mechanical data to modify further the geometry and/or assigns mechanical data to areas of the idealised domain. This step is heavily application dependent and is still under development.

During the first step, the constraint set on this algorithm is obviously that the process must respect the initial geometry, i.e. initial and idealised geometry must be consistent. Geometrical criteria are used here to guarantee that the final state of the model will not be geometrically incoherent with respect to the initial state. These criteria are based on the thickness, the curvature and the size of the idealised area.

During these processes, BC data, B-Rep topology and form feature attributes must be transferred on the adapted model. Finally, the FE mesh model is generated from the adapted model and enriched with the proper mechanical parameters available in that model (see level © of Fig. 15). As a result, the proposed environment at Figure 15 (B) can be integrated into an industrial process without modifying the pre-existing CAD and FE mesh generators since mesh generators often accept polyhedral model as input.

In the case where form features are available on the polyhedral model, we add a new category of details named simplification features defined in the next section. Such a type of details is not removed automatically, because their identification depends of the order of application of the simplification operators and of the phenomenon under consideration. For example when a simplification feature is identified before skin details, then new simplification features may appear after the application of the skin operator. However, during the simplification process the topology of the form features, of the BCs, B-Rep topology may evolve.

¶

#### 4.1.2. Tessellation process of form features model:

Because the tessellation process is performed patch by patch on the B-Rep model, the From feature model can be tessellated for each form features, if the Form feature graph (SFOG) or other source such STEP data, is available at B-Rep level, such these type of semantic allow to user to specify exactly the parameters of map of sizes, without affect other form features.

The figure 19 shown such process based on the feature recognition process, and the figure 20 is an example implemented on the tessellation environment when the form feature is available

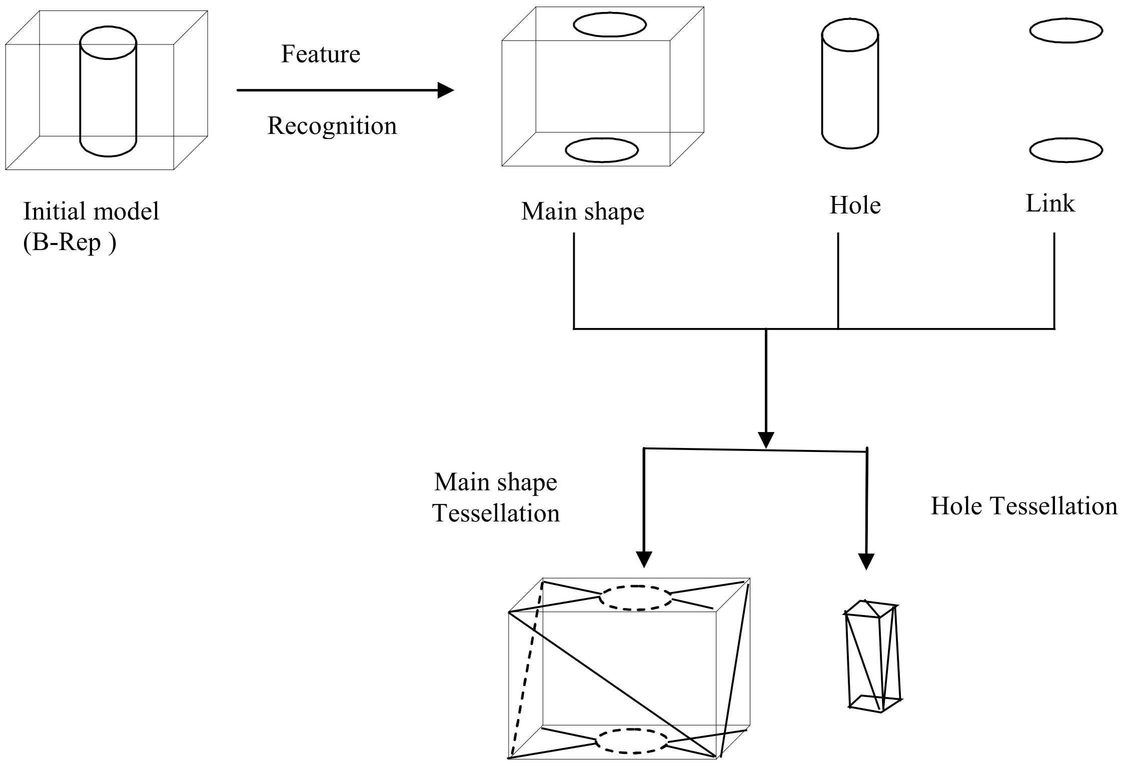



**Figure 19** Tessellation of form feature model

at B-Rep level, the user select the desired form feature from Feature model, and then tessellated it, after can be applied the local map sizes which corresponding to the parameters of form feature (like radius of holes, correspond to radius of map of sizes) in order to simplify the form features.

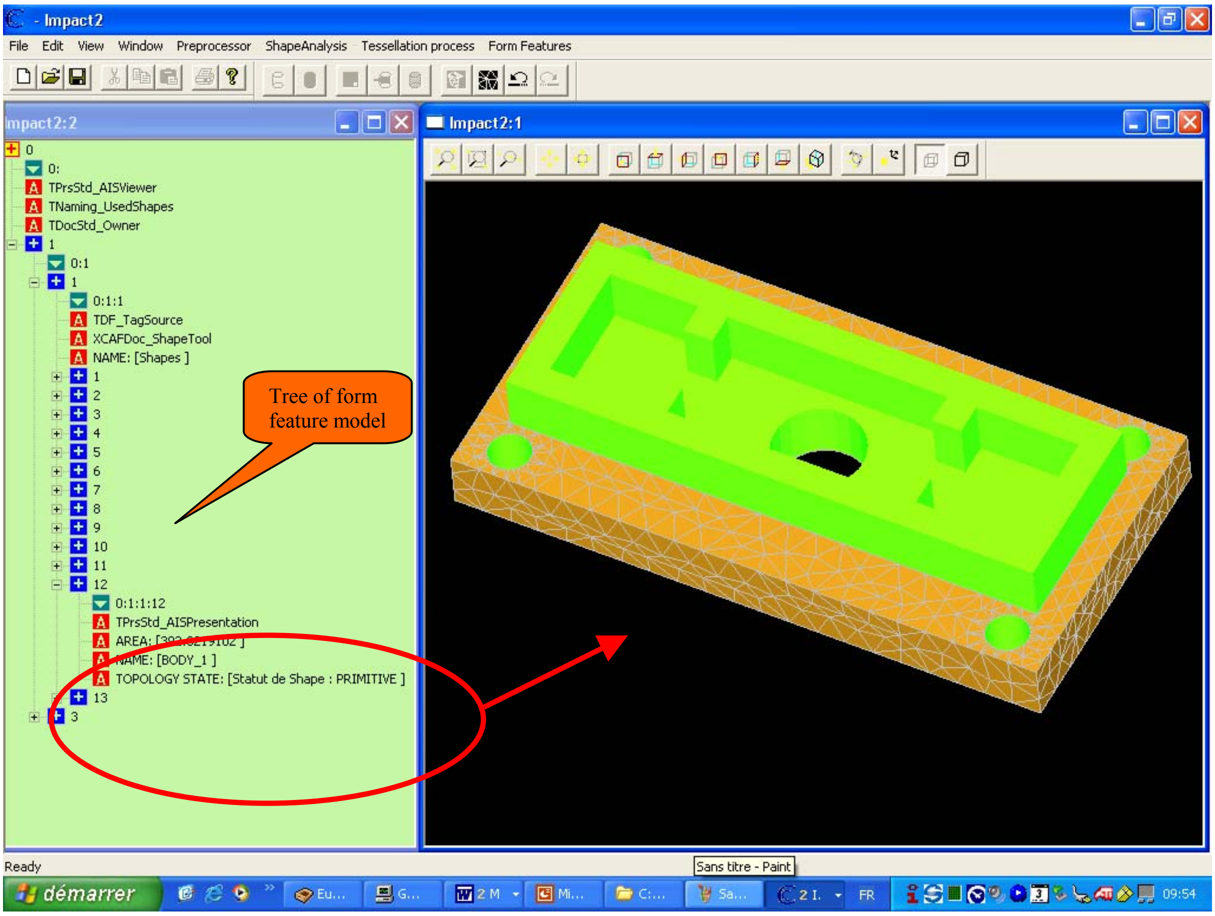



**Figure 20** Selection of one form feature from form tree of  feature model for tessellation process

### 4.1.3. Simplification process from feature model:

¶The simplification of the form features directly from model feature model make the problem of validity of B-Rep model, when the removal of a form feature implies the changes of topology and geometry of the original model, thus the interpretation of form features after simplification.¶

¶It is for what one we choice the polyhedral model like intermediate model for the simplification of the form features.¶

¶The graph of the Form feature (SFOG) model is initially transferred onto the B-Rep model, and then it transferred onto the polyhedral model, in this case the set of polyhedron faces corresponding to the features can be removed locally, also the contour associated to the form feature (like hole) can be remeshed locally. This process can be done a priori on the tessellation environment (23, 24) if the feature information does not present on the B-Rep

model, otherwise the B-Rep semantic like (Cylindrical face, Plane face, .),may be sufficient to recognise certain form features on the polyhedral model.
Then we purpose this schema for semantic propagation of form feature and B-Rep:
Feature model ◀——▶ B-Rep model ◀——▶ Polyhedron model
¶

**Figure 22** Tessellated model with Holes (edge length =5)

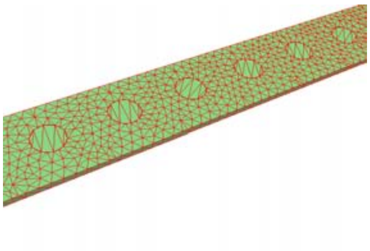


**Figure 21** Topological simplification (a posterior recognition)

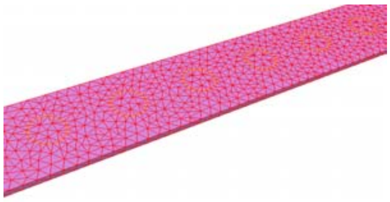


**Figure 23** Tessellated model without Holes (a priori recognition of holes ).

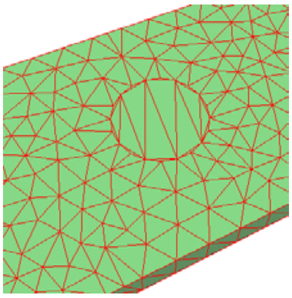


**Figure 25** Local remeshing operators (Simpoly)

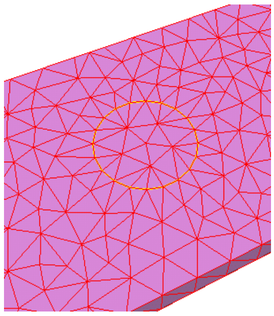


**Figure 24** Tessellated model without Holes (a priori recongition of holes).

### 4.1.4. Simplification features

In our context, we define a simplification feature as a form feature whose removal doesn't affect the analysis results. This is based on input analysis parameters, like the map of sizes, expressed a priori by the user. Therefore, the fact that a form feature is also a simplification one is governed by the mechanical configuration under evaluation, which is represented by the map of sizes. Then, broadly speaking, a feature defined on the polyhedron model represents a simplification feature, if it can be considered a detail, as described in section 4, that means that the map of sizes associated to the feature fully contains it.

As such, a simplification feature aggregates geometric as well as mechanical data:

- At least, a connected set $F$ of geometric elements constituting one or more form features, and possibly parts of features which are adjacent to them in the feature relation graph,
- A set of mechanical data characterised at least by a map of FE sizes to reflect, a priori, the user's view of the discretization of the structure in accordance to the objectives of the

simulation. Other data concerning the BCs, the constitutive material, ..., could be also useful,

in a way such that one among the three categories of details described in section 4 can lead to the corresponding removal or abstraction processes to effectively modify the shape of the component by removing or abstracting *F*. Indeed, this means the FE sizes attached to the feature geometry are 'large enough' to change totally the shape of *F*. If *F* is reduced to only one form feature and if the map of FE sizes allows only a partial shape transformation of the feature, this transformation can take place using the corresponding operator but simplification feature processing cannot take place.

It should be noticed that the set *F* mentioned here above designates indeed the polyhedral representation of the form features lying in the CAD model. This polyhedral representation contains the form feature data available from the CAD model. In the examples shown in figure 6, configuration © is not a simplification feature due to the dimension of the discrete envelope associated to the blind hole $h_1$ in configuration (a), while the hole $h_2$ is a simplification feature in configuration (d) where the discrete envelope gets larger. For model analysis preparation, not all the form features may either have the same impact or be useful, some of them provide more hints for simplification than others. Similarly, different simplification features can be associated to or may contain the different categories of details described at section 4 depending on the content of the set *F*.

In general, for form features generating internal depressions, e.g. holes, pockets and slots, being a simplification feature depends on the ratio of their dimensions with respect to that of the map of FE sizes. This reflects the fact that for being removed without altering the analysis results, not only the feature faces have to be fully included in the related map of FE sizes but also their associated virtual volume, i.e. the volume that must be subtracted to the object when the feature is inserted. In case of through holes having available simplification features data, these data can definitely speed up the simplification process compared to the complexity of the topological operators. Whereas for form features not changing the genus of the object, their removal requires the specification of new ad hoc removal operators on the tessellated model, and the process is probably not so much more efficient that simply applying the skin detail removal process.

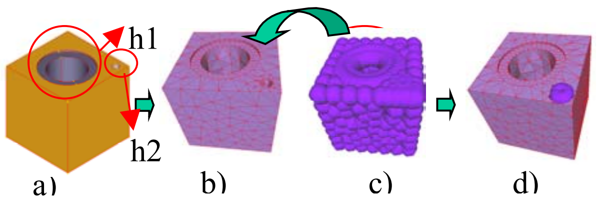



**Figure 6: (a) initial CAD model with two blind holes (radius h1 = 25, radius h2 = 5), (b) polyhedron model (c) map of FE sizes with spheres of radius 5, (d) h 2 is a simplification feature: the map of sizes fully contains it.**

Fillets and chamfers form features can be as well very useful and associated to special skin detail removal operations depending on the analyst's requirements, the type of FE used, ... In particular, information about the type of the adjacent faces can highlight details, whose removal can originate different configurations according to the simulation to be performed and other expert decisions concerning the preparation phase. As described at section 4.1, the information about the type of surface associated to a B-Rep face originates from the STEP file input from standard CAD or feature modellers. At this stage, such high level information concerning primitive surfaces for B-Rep faces adjacent to chamfers or blends is clearly the critical information to characterize a simplification feature rather than the nature of the connecting surface, i.e. the blend or chamfer, …, which in any case is a detail for the analysis specified.

In figure 6, the simplification feature associated to the fillet having adjacent faces forming a 90° angle is shown, with two possible associated removal operations:

- Extension of the adjacent faces (b),
- Substitution with a planar face ©.

In both cases the new configuration to be applicable should be inside the given map of FE sizes. These two configurations are useful complements to the skin detail operator to better fit the user’s requirements.

Finally, ribs can be related to abstraction details to express a local structural behaviour of either plate or shell type. Here, information on parallel faces can be automatically derived, and if the corresponding volume describing the set *F* is fully contained in the associated map of FE sizes, the rib is candidate for being abstracted. Again, this example shows that simplification feature data can speed up and ease the abstraction process compared to the basic treatments described at section 4. Simplification features are complementary to the skin, topological, abstraction operators described at section 4.

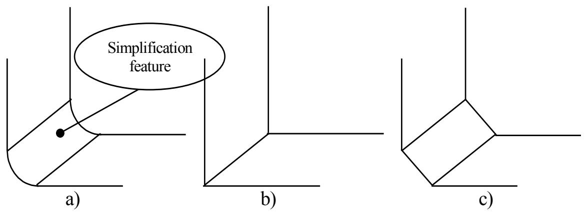



**Figure 26** Fillet (a) and associated simplifications (b, c) depending on user’s expertise and simulation objectives.

### 4.1.5. Idealization process using the feature approach:

Currently the idealization process is based of the low topological entities (vertex, edge, faces), in order to localise the idealized areas; this process is based in tree algorithm in order to detect the opposed surfaces of the model (see Figure 27A). This algorithm create two groups of faces, representing two opposed surfaces of the model. If these two surfaces satisfy the geometrical criteria - i.e. they are parallel, within a certain distance (see Figure 27b), with constraints on their curvature and size -, then they define a portion of the volume of the part which can be idealized.
The new approach proposed by using the features information on the initial CAD model (B-Rep representation) presents a complement of the first approach in order to propagate several information about the idealization process, such the parallelism between surfaces, relations between surfaces, volume, area, thickness.

Currently this process is semi-automatic when the 'discrete envelope' is used to characterize idealizable areas.

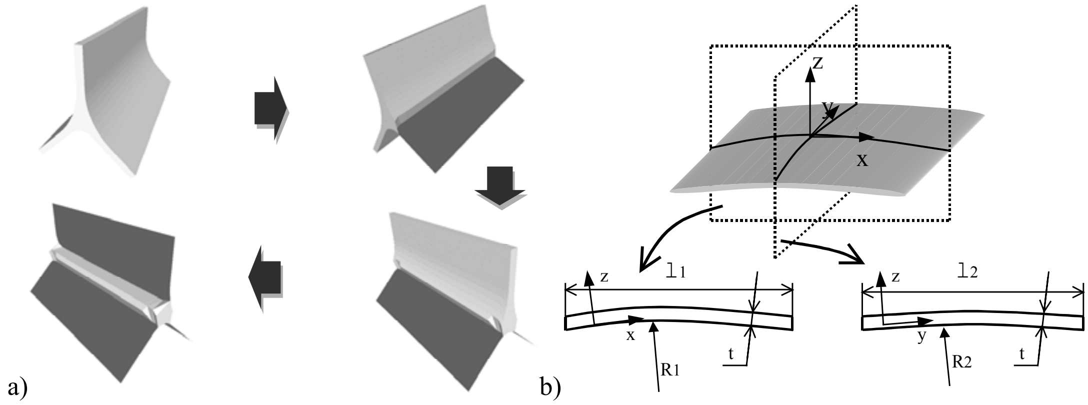



**Figure 27 A: Idealisation process a) continues process, b) geometrical criteria for shell or plate idealisations**

By feature approach the current process can be fast, when a form feature information are transferred on the B-Rep model and after on the polyhedron model, for example the information like parallelism between two topological faces is important for characterize the idealisable areas, for example figure 28,
This information is maintaining in the SFOG graph (see figure 27).

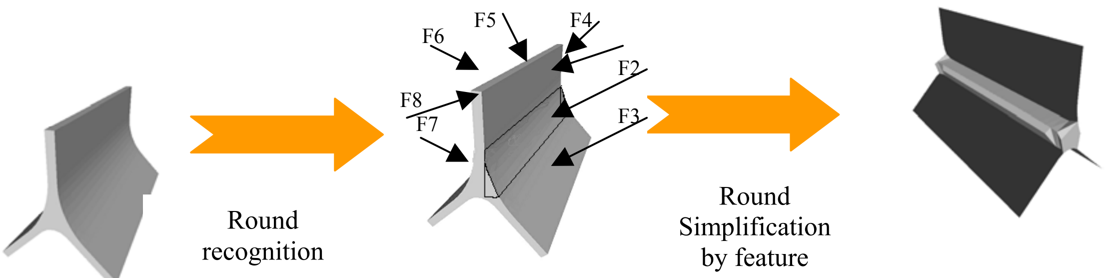



Figure 24a Figure 25a Figure 26a

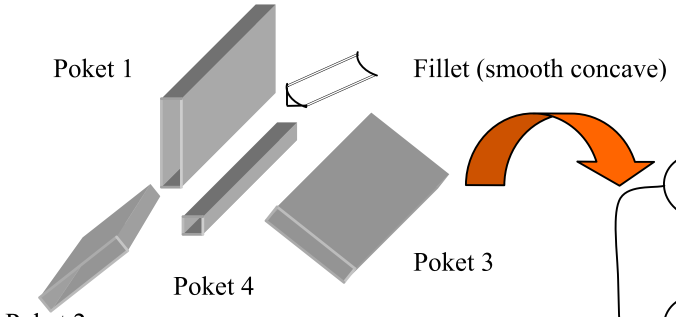



Figure 27a

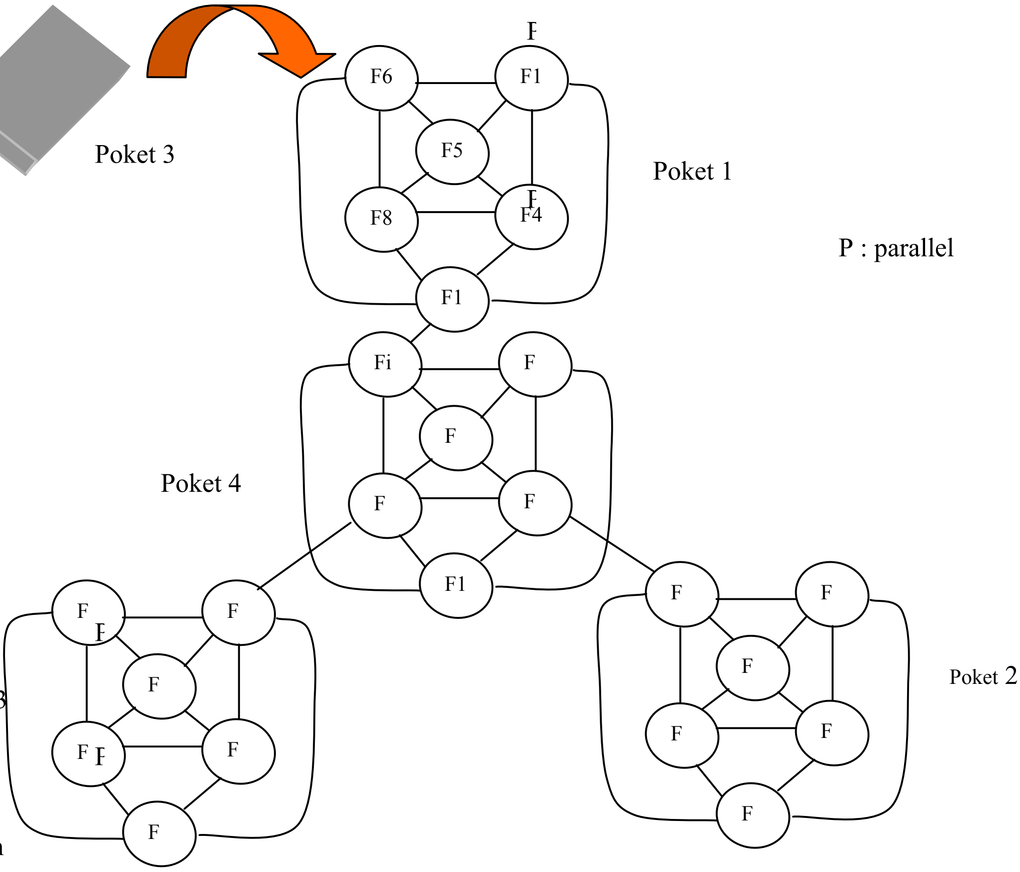



Figure 27 : FFGraph

Technology of form feature recognition and extraction is deemed as a fundamental technology for abstraction of an analysis model.

The first step of proposed approach is to recognize the feature from the solid model (boundary representation), in order to reduce the complexity of the simplification task from polyhedron model.

The recognition process involves the classification of geometric entities and identification of geometric relationships between adjacent features.

Applying the principle of feature recognition and extraction [10, 11], removal of detail features
( Through-hole, blind hole, Rib ,boss, blind slot, round, fillet,..) depend in the objective of the simulation and context of analysis ( structural analysis, thermal analysis,…).
In order to identify this details the **map of sizes** criteria must be introduced (by the user ) to compare locally between the characteristic dimensions of each feature (feature parameters such round of hole), and the sizes of the elements (size edges target ) who have to be used for mesh generation stage.
This comparison we allow to put in evidence which have no significant influence on the physics phenomena.
Indeed the map of sizes can be defined a prior by the user by using the knowledge of the user and the feature information (round and position of the hole), or a posterior by using error criteria based on the analysis results which is already performed.

### *Needs of feature information from solid model?*

Associate feature information on the polyhedral model, the user select feature witch will be removal. This suppression is performed by the map of sizes defined locally by the user.
The information of feature we allow to specify the dimension of this map of sizes in order to simplify the details without affecting the other details.
For example the hole suppression can be performed, by the localisation of this hole on the polyhedral model and his associated parameters (round, ..), in order to remove it.
The radius of the hole defined such as the radius of the error zones which defined the map of size.
The next stage is to generate the polyhedral models which represent the intermediate model between the solid model (feature based representation) and the simulation model.
The second step of the simplification process is based primarily on the simplification of the topological entities (faces, edges, vertices).
Also we can take an advantage of the first process of recognition by propagate the information about the features in order to characterize idealizable areas, or boundary conditions.
Currently the idealization process is based of the low topological entities (vertex, edge, faces), in order to localise the idealized areas; this process is based in tree algorithm in order to detect the opposed surfaces of the model (see Figure 28 a). This algorithm create two groups of faces, representing two opposed surfaces of the model. If these two surfaces satisfy the geometrical criteria - i.e. they are parallel, within a certain distance (see Figure 24), with constraints on their curvature and size -, then they define a portion of the volume of the part which can be idealized.
The new approach proposed by using the features information on the CAD model (B-Rep representation) presents a complement of the first approach, because several information about idealization process can be propagate for characterize idealizable areas, such the parallelism between surfaces, relations between surfaces, volume, area, thickness.
Currently this process is semi-automatic when the ‘discrete envelope’ is used to characterize idealizable areas (see figure 24a, 25a, 26 a).
This information is transported via the SFOG (Shape Feature Object Graph). This graph is keep and update throughout all process for analysis model creation

This approach which allows the taking into account of the initial geometrical model, can be to combine upstream in the simplification process (decimation and idealization), before the generation process of the polyhedral model, this led to a taking of very significant time, at the same time enables us to keep the tractability of the adaptation process either a priori or a posterior.

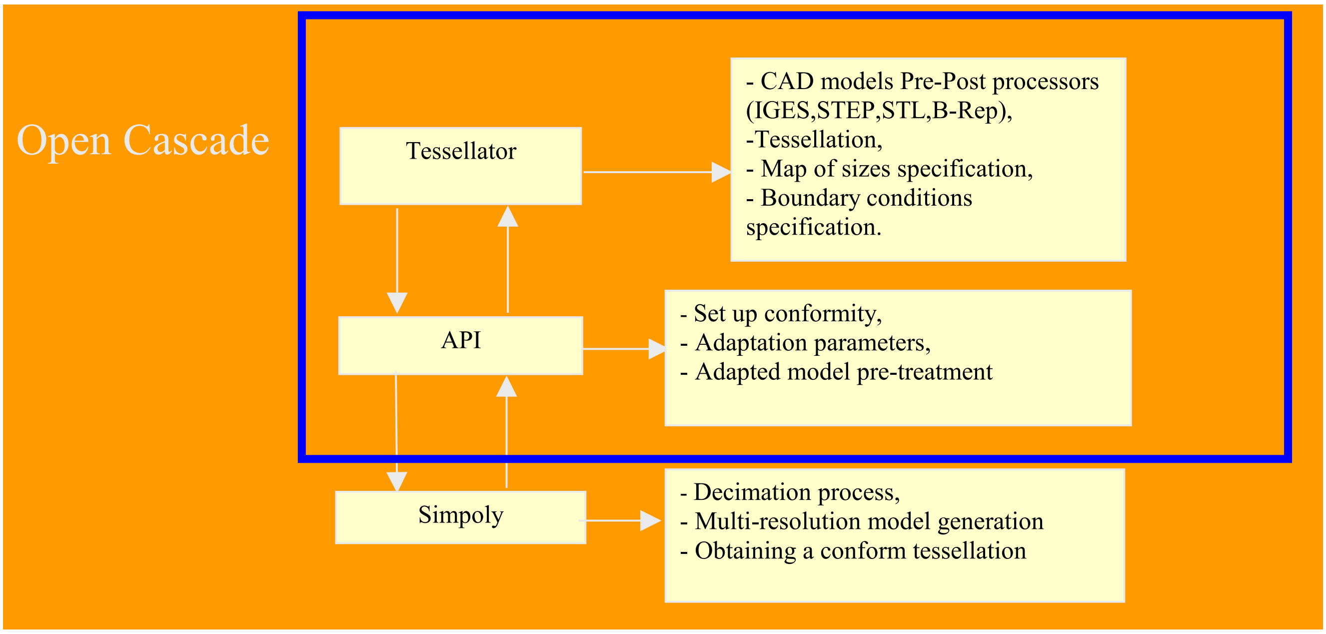



**Figure 28** The previous implemented data structure

The coherence between a solid model and a simulation model is achieved if we maintain the coherence between a solid model and the tessellated model (polyhedral model).

## 5. Identification of information needed for analysis model preparation

The objectives of my thesis are:

- To reduce the task of model preparation using the Form Feature model as height level model,
- Identified certain of the attributes which are important to generate an analysis model,
- To find the relation between this attribute in order to avoid the duplication of information,
- To Maintain the coherence between the data structures of the CAD model and simulation model,
- Propagate the form features attributes and B-Rep attributes through all the processes of analysis model preparation,
- To identify the method to localize the modification area on the polyhedron model, in order to avoid to regenerate the polyhedral model, and to compare locally the result of the both models (initial, modified): impact evaluation.

Two solution exist for analysis model generation:

**Solution 1** steps to be done are:

1. Design (possibly feature-based OR Brep + recognition)
2. Tessellation
3. Set-up conformity
4. Domain of study set up
   4.1. Boundary conditions set up
   4.2. Restriction of the part of the model to study
5. Simplification
   5.1. Details identification
   5.2. Simplification: geometrical, topological, and dimensional.
6. Idealization

**Solution 2** steps to be done are:

1. Design (possibly feature-based or Brep + recognition)
2. Domain of study set up
   2.1. Boundary conditions set up
   2.2. Restriction of the part of the model to study
3. Tessellation
4. Set-up conformity
5. Simplification
   5.1. Details identification
   5.2. Simplification: geometrical, topological, and dimensional
6. Idealization

All the information needed for the both solutions is represented in the table 1.

| **PROCESS** | **Entity needed** | **Information needed** | **Information useful for speeding up the process or for maintaining information**[1] | **Output model** | **Type of Information from the input model or generated to be propagated to other processes** | **Process to which it has to be propagated** |
|---|---|---|---|---|---|---|
| Domain of study set up : **Boundary conditions set-up** | - volume<br>- faces<br>- portion of faces<br>- edges[2]<br>- portion of edges<br>- vertices | - face element<br>- edge element<br>- coordinate | | If Solution1 Brep<br>If Solution2 tessellated model | presence of a boundary condition | - tessellation<br>- conformity set-up<br>- simplification<br>- idealization |
| Domain of study set up : **Restriction of the object to study** | Features Brep -like entities[3] | - object symmetry<br>- features<br>- features distance<br>- face parallelism<br>- face distance<br>- parallel. | | If Solution1 Brep<br>If Solution2 tessellated model | Symmetry cutting planes | - tessellation<br>- conformity set-up<br>- simplification<br>- idealisation |

[1] i.e. information that could be propagated from the previous steps BUT not present or used now
[2] it could also be a curve on a face, thus not belonging to the Brep

| | | | | | | |
|---|---|---|---|---|---|---|
| | | betw. edges | | | | |
| **Tessellation** | faces<br><br>edges | - parametric description of the face surface<br>- type of face geometry<br>- parametric description of edges<br>- coordinates of edge end points<br><br>- edge length | - points on surfaces corresponding to boundary conditions definition, e.g. curve points internal to a face[4] | tessellated model[5] with some triangles having only two adjacent triangles and duplicated nodes corresponding to those of border | - (features )<br>- face information [6]<br>- edge information [7]<br>+<br>- symmetry cutting planes<br>- boundary conditions | - conformity set-up<br>- simplification<br>- idealization |
| **Conformity set-up** | triangles<br>points | - | - polyedges<br>- boundary conditions | connected triangulation with all triangles adjacent to exactly 3 triangles for manifold objects[8] | - (features )<br>- face information<br>- edge information<br>- symmetry cutting planes<br>- boundary conditions | - simplification<br>- idealization |
| **Simplification Details identification** | - nodes | - map of size | - features<br>- (poly)faces<br>- polyedges<br>- boundary conditions[9] | not changed - labeled | | |
| **Simplification: geometrical, topological, and dimensional** | - node<br>- partitions | - | - features<br>- (poly)faces<br>- polyedges<br>- boundary conditions[10] | updated triangulation | - (features )<br>- face information<br>- edge information<br>- symmetry cutting planes<br>- boundary conditions | - simplification (loop)<br>- idealization |
| **Idealisation** | - node<br>- triangles | - parallelity among polyfaces | - features<br>- (poly)faces<br>- polyedges | - non-manifold model | - | |

[3] in case it is performed on tessellated model the same content of information of Brep should be provided, i.e. collection of triangles in faces, edges.......
[4] these points should belong to the generated tessellation
[5] Now it is a unique polyhedron, but adding the concept of partition (polyface) we could group all the triangles obtained for each face, adding also the concept of polyedge (partition on edges of the tessellation) could speed up the process for conformity set-up because the evaluation of the point distance could be performed only on the nodes belonging to the polyedge structure
[6] i.e. grouping of constituting triangles + type of surface
[7] i.e. sequence of edges (or nodes this **has to be defined**) of the triangulation belonging to the same curve
[8] this requires the updating of the polyedge structure defined during the tessellation process, in which possibly two polyedge have been created for an edge of the Brep, i.e. one for each adjacent face, after the conformity set up, only one polyedge for edge of the Brep should be present
Polyedge could be used also for the specification of curves internal to faces on which Boundary conditions have been defined
[9] for putting conditions on the simplification process, eg. points on the symmetry planes
[10] for putting conditions on the simplification process, eg. points on the symmetry planes

| | | | | | | |
|---|---|---|---|---|---|---|
| | | - material | - boundary conditions<br>- parallelity among polyfaces<br>- distanc. betw. features<br>- feature parameters | | | |

Table 1: Information needed for model preparation

# 5. Overview of the high level topology (HLT) based environement

The HLT structures used to support the analysis model preparation, the tessellation of the model and their interactions are central to the analysis framework. The HLT topology model and attributes are used to house the problem definition. The general nature of the attribute structures allow them to also be used for defining numerical analysis attributes.
The general interactions between the components are shown in Figure 29. These interactions are described in more detail in the following chapters, with the remainder of this chapter introducing the basic concepts of the three structures.

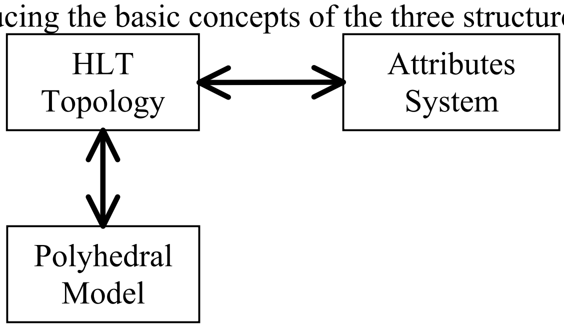
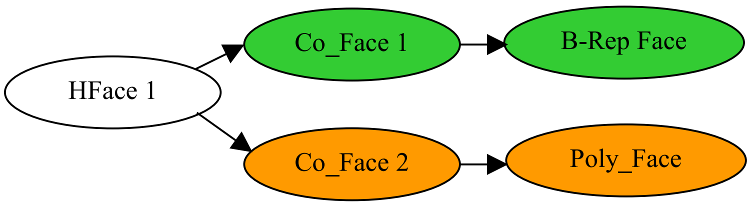



**Figure 29** Relationships between the component of HLT based environment.

## *5.1 The high level topology*

The topological representation used is based on the radial edge data structure of Weiler, the topological hierarchy and the relations between the entities is shown in figure 31.
The topological entities of Vertex, Edge, Loop, Face, Shell and Region are sufficient to give an understanding of the topology in the case of 2-manifold models. However to fully understand the topology in the case of non-manifold models it is necessary to have additional information. This additional information is in the form of HCo_entity which describe the connection of one entity to another.
The simplest way to think of HCo_Face is to consider a face. Each entity has two sides, each of which may be attached to a region. Thus, the face is said to have two HCo_Face, one associated with each side. Each HCo_Face has one face as children entity, or set of edges, or an isolated vertex. As with a face, each loop has two uses, one on each side of the face associated with the loop.

Note that it is really the HCo_entity that define the topological connections between the various entities as shown in Figure 31. The other topological entities: HRegions, HFaces, HEdges, and HVertices connect sets of uses together and provide the shape information that turns the model from a purely topological object into a geometric object. Even though the basic topology is given in terms of the use entities, it certainly is meaningful to discuss things like the “set of edges bounding a face”, since this is a relation that is derived from the use entities.



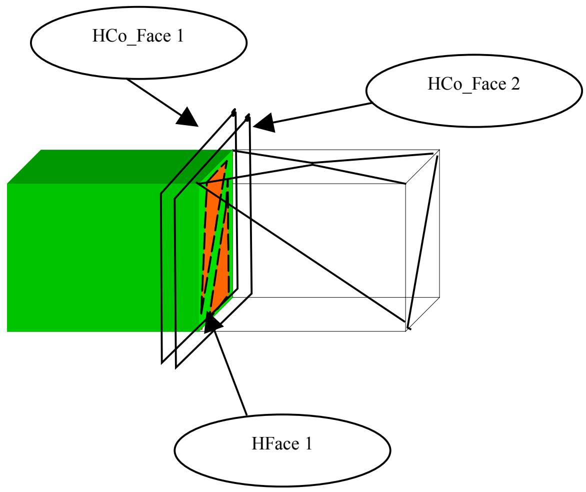




**Figure 30** An example HCo_Face-principle

The main advantages of the additional information, a such Co_Face for example is to be able to combine several representation in the same entity called high level entity, which allow to maintain the consistency between several representation (like B-Rep and polyhedral representations) see figure 30.

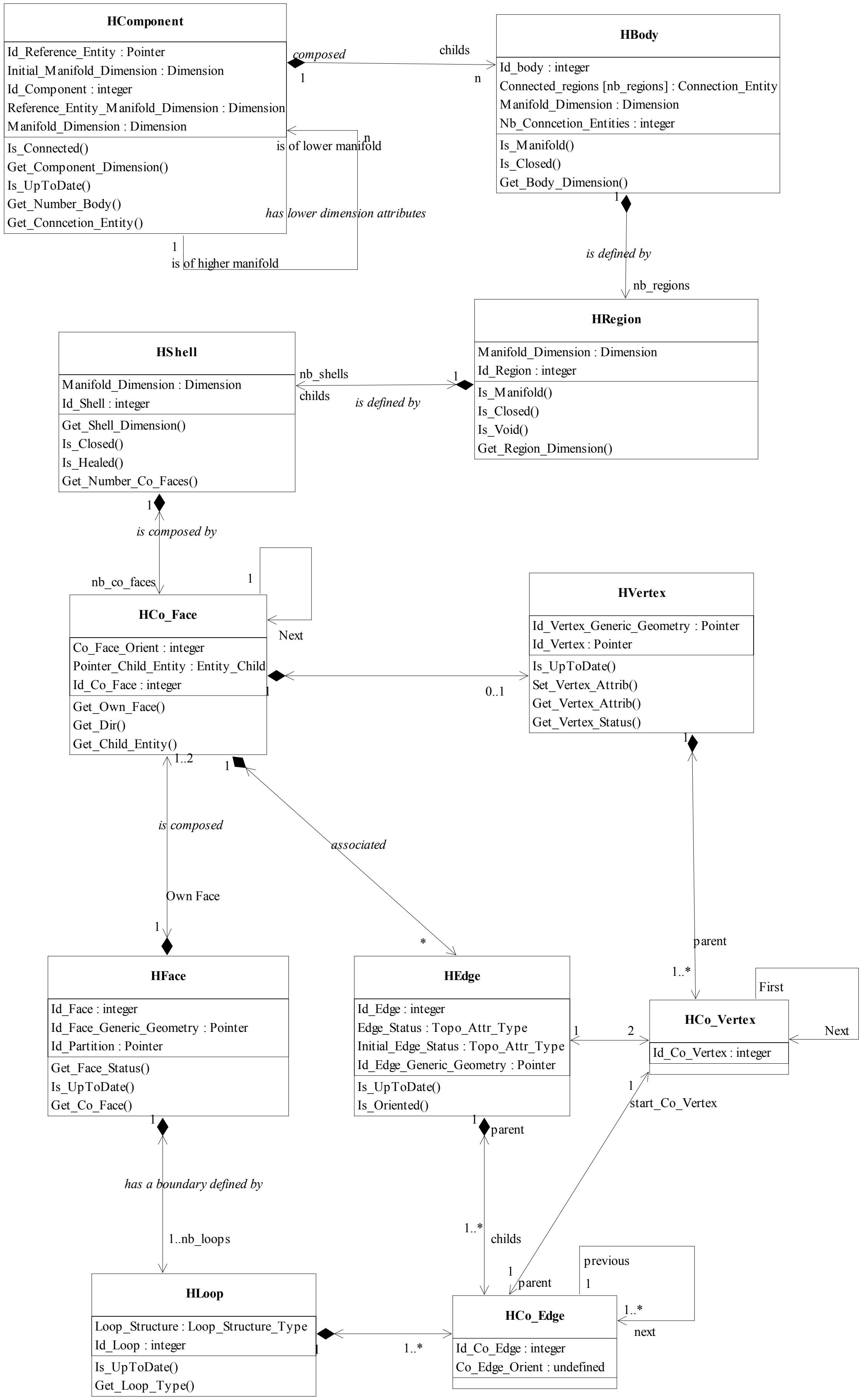



**Figure 31** Topological representation of HLT in UML modeler

## *5.2 Topological entities:*

The topological entities in HLT representation are classified into three categories:

**Grouping entities**: HComponent, HBody, HShell, HLoop.
**Basic topological entities** : HRegion, HFace, HEdge, HVertex.
**Virtual entities:** HCo_Face, HCo_Edge, HCo_Vertex.
The virtual entity is added in order to characterize non-manifold configuration along respectively the face, edge, and vertex.

### 5.1.1 Grouping entities:

**HComponent (High level component)::** one element of an assembly that has an individual behaviour with respect to the perception of the user in the context of a given application: digital mock-up, structural analysis, VR ergonomic application, ...

We added to this definition a set of topological properties:

- *Initial_Manifold_Dimension*: This parameter is equivalent to the Manifold_Dimension parameter. It represents the initial value of this parameter prior to the application of any shape changing operator that could result in some 'idealization' of the initial shape.
- *Reference_Entity_Manifold_Dimension:* In the context of the application set up a B-Rep Entity either can represent directly the target object or the location of attributes onto a target object. In the latter case, the location of the attributes follows the same principle as any other B-Rep Entity. However, these attributes are located into another Topological entity whose manifold dimension is necessarily higher than or equal to the Manifold_Dimension of the B-Rep Entity describing the attributes. As an example, a surface pressure area applied onto a Volume is assigned a Manifold_Dimension of 2 and a Reference_Entity_Manifold_Dimension of 2 since it is applied on a Surface which is topologicaly described as a Face.

Similarly, a force applied at a Point on to a Volume structure is assigned a Manifold_Dimension of 0 and a Reference_Entity_Manifold_Dimension of 2 since it is applied on a Surface.

- *Manifold_Dimension:* The number of independent parameters required to describe a given entity from a B-Rep point of view. This parameter ranges from 0 (a Point) to 3 (a Volume). Once the Manifold_Dimension of an Entity is specified, the B-Rep approach used here means that NO Entity of Manifold_Dimension GREATER or EQUAL to that of the Entity described must take part to the description of the current entity. This is the basic principle of the B-Rep approach.

- **HBody (High level body):** A Component must be composed of at least one body and may be formed by more than one. Is this case the bodies must be disconnected in the sense of connectivity topological property. Two distinct bodies cannot share Faces, Edges or Vertices otherwise they would form a unique non-manifold body.

Two distinct bodies from the topological point of view must be distinct and non interpenetrating from the geometric point of view.

- **HShell(High level shell):** A Hshell is a set of HCo_Face that form a closed boundary. Each shell is associated with zero or one regions. A shell will not have a region associated with it if the HCo_Face that make it up have no region associated with them.
- **HLoop (High level loop):** A HLoop is an ordered collection of HCo_Edge that form a closed loop. Generally, loops are closed, having no actual start or end point, but they may be open. Three types of loop: boundary loop, contact loop, hole loop.

### 5.1.2 Basic topological entities:

- **HVertex**: A vertex occupies a single Point in space. A vertex can be used to bound a curve and to specify a location for a node. A vertex is a 0 (zero) dimensional manifold. A vertex which is located in the interior of a Face is called a key point. It is used to force a point to be located at that specific geometric position.

- **HEdge:** The topology associated with a curve or a segment (an edge of a polyhedron) or a polyline (a sequence of edges of a polyhedron). An Edge is a 1 (one) dimensional manifold. Edges are bounded by Vertices. One Edge is associated to exactly one Curve (segment or other type of Curve).

- **HFace:** A connected portion of a surface or a set of facets (in case of polyhedron) bounded by a one or more Loops of Edges. A Face is a two dimensional manifold.

- **HRegion:** A region is a 3-d topological entity bounded by the set of shells, in general, a region seems to be a bounded volume described by a shell. However, HCo_Faces may point to single vertices or edges an shells are just collections of HCo_Faces. Further, a shell need not consist of a closed set of HCo _Face. So, it is possible that a region does not represent a volume at all. We added the closure property to this definition in order to distinguish a material-filled region from an empty one.

### 5.1.3 Virtual entities:

- **HCo_Face:** A Co_Face denotes the appearance of a face in a solid; in the manifold case each face should have one Co_face with inside orientation of the region, for the non-manifold case, each face should have more than one Co_Face. This concept is similar to the Face_used introduced by Weiler, but in our representation the Co_Face do not include any boundaries of faces, which is case of Weiler data structure.
  In a non-manifold model, a face bounds two incident regions, and thus each side of a face should be a part of the boundary of each region. To meet this requirement, we split the face into two Co_Faces. A set of Co_Face compose a shell. The Co_Face can be assoiated with not only a face but also a wire edge or isolated vertex in a region. The Co_Face is used to characterise the non_manifold conditions where a face is adjacent to two regions.
- **HCo_Edge:** A Co_Edge is a unique Entity set between two Vertices. This Entity expresses the use of Edge in a Loop defining a Face. One Edge is represented by several Co_Edges. In case of an Edge located at the boundary of two Faces, this Edge will be represented by exactly two Co_Edges. In should be noted that the Co_Edges

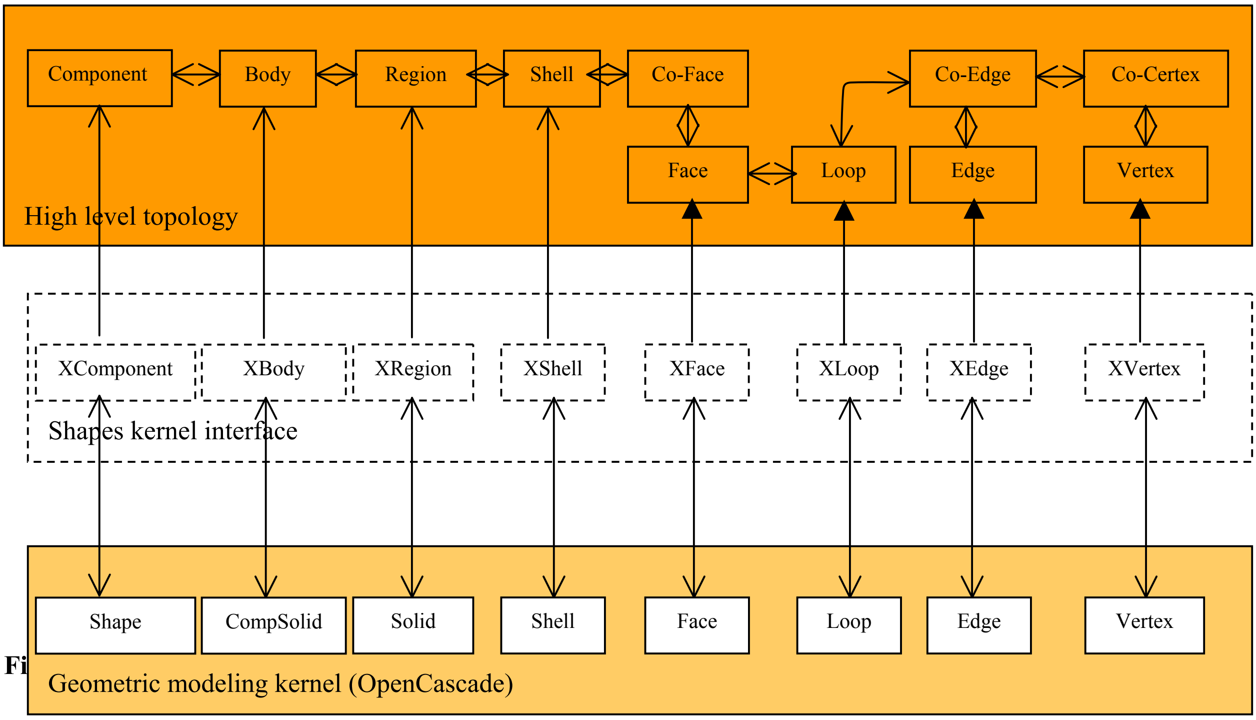
will be oriented in both ways to characterize the orientation of the coresponding Face. These rules fulfill in this case the Moebius Surface orientation principle. In case of a non manifold face, more than two Co_Edges can be attached to a same Edge.

- **Co_Vertex :** In a non_manifold model, a vertex can be adjacent to an arbitrary number of two manifold surfaces. Note: a two_manifold surface is formed by a group of connected faces, and wire edge can be dealt with as a degenerate case of surface. We introduce the Co_Vertex in order to handle such a no_manifold condition at a vertex.

### *5.3. Model interfaces*

The model related classes are designed to be wrappers around functionality that is provided by an underlying geometric modeling kernel. The reasons for using a geometric modeling kernel, rather than directly implementing geometric calculations in the model entity classes, are consistency and simplicity. A modeling kernel constructs a model using a certain set of algorithms and tolerances, if one attempts to use different algorithms or tolerances when interpreting the model information it is quite possible to get slightly different answers that lead to inconsistencies between the original model representation and the current representation. Also the task of constructing a complete geometric modeling system is a huge one, thus it makes sense to leverage the work that has been done by others in this area.
The reason for using the model abstraction presented in this chapter rather than directly querying various modeling kernels for the geometric information is to provide a consistent representation of the model regardless of the underlying kernel implementation. Even though all of the major modeling kernels provide a boundary representation of the model, they all have differences in how they represent that topology. To expose all of these differences to the rest of the geometry-based environment would greatly complicate the system. By providing a consistent interface, the rest of the system is insulated from these differences which are all encapsulated in the model interface classes.
The modeling kernel must provide two main types of functionality:

- Extraction of information about the entities in the model and their topological relations as




Fi

needed to build up the HLT representation described in section 5.1,

- Functionality to answer the topological queries that are present in the Shell, Face, Edge and Vertex classes.

In generally at a minimum a interface to a modeler will consist of five classes, one each derived from the base classes Region, Shell, Face, Edge, Vertex, as shown in Figure 26 using the Open Cascade Shapes interface as an example. The collection of these classes is referred to as the Shapes kernel interface.
Abstract topological data structure in Open Cascade describes a basic entity, the shape, which can be divided into the following component topologies:

- COMPOUND: A group of any of the shapes below.
- COMPSOLID: A set of solids connected by their faces. This expands the notions of WIRE and SHELL to solids.
- SOLID: A part of 3D space bounded by shells.
- SHELL: A set of faces connected by some of the edges of their wire boundaries. A shell can be open or closed.
- FACE: Part of a plane (in 2D geometry) or a surface (in 3D geometry) bounded by a closed wire. Its geometry is constrained (trimmed) by contours.
- WIRE: A sequence of edges connected by their vertices. It can be open or closed depending on whether the edges are linked or not.
- EDGE: A single dimensional shape corresponding to a curve, and bound by a vertex at each extremity.
- VERTEX: A zero-dimensional shape corresponding to a point in geometry.

Each of the derived entity classes (XShape, XBody, XRegion, XShell, XFace,XLoop, XEdge and XVertex in Figure 32) must override a minimum set of functions to provide the needed functionality. These are the member functions most part these are geometric query operations.

The main responsibility of the derived model class (XShape in Figure 32) is to extract the topology of the model and create the derived entity classes for each entity in the model. In doing so, the topological representation of the model is set up correctly by the base entity classes.

# 6. Attribute system for analysis model preparation

Given our current automated simulation technologies, the key step to their integration with an enterprise level system is to be able to maintain all problem definition information within the product data management system. Since our automated mesh generation tools already operate from the same solid model representations, the domain definition piece of the problem definition is already taken care of. Although it may be possible to employ the attribute specification capabilities of those solid modelers to specify the analysis attributes, their inability to support the forms and functions needed by analysis attributes makes them unattractive. Therefore, it is desirable to build on the type of analysis attribute specification capability described in this section. The problem with the available implementation is they used an independent set of data structures.

To support the effective specification of attributes for the complete set of related analyses, while at the same time making it efficient to collect the attributes required for each specific analysis, an organizational structure is needed for the purpose of describing sets of attributes. The organizational structure must effectively support a design process for scenarios where multiple physical behaviors must be evaluated. In many cases, the result of one analysis represents part of the problem definition of another. For example, consider the situation of performing thermal, electrical, and thermal-mechanical analyses of an electrical component. Though the three analyses are quite different, there is an overlap of attribute information. The base materials are the same for all three analysis types, while the boundary conditions and loading conditions vary among the three. The thermal analysis would study various thermal load distributions. The thermal-mechanical analysis would use the resulting temperature fields as input to its load cases. The ability to effectively organize hierarchies of attributes as needed for each of these analyses is critical to a useful attribute management system.

## *6.1. Attribute Classification*

A classification system for attributes depends on a number of factors, prime among them are the domain of application and the type of information the attributes represents.

According to the needs discussed on the previous sections:

- Boundary conditions specifications,
- Transfers of form features information onto B-Rep model and Polyhedral model,
- Maintain the attributes during the analysis model preparation (Tessellation process, set up conformity, simplification process),
- Check the validity of attributes during the analysis model preparation.

### 6.1.1. Topological attributes:

These are vertex, edge, face, loops, shell, or region attributes.

Any attribute attached to a vertex entity is known as a vertex level attribute. For example, a vertex may be assigned attributes by type of vertex like boundary vertex, isolated vertex, contact vertex, surface vertex, homologue vertex, or feature vertex.

A face attribute can describe the geometry type of face. For example, planar, cylindrical, B-Spline,.. etc. A face attribute can also assigned attributes by the type of face like boundary, isolated, contact, ..
A region attribute can describe the manifold dimension of region, or state of region like closed region or not, the closure information it is important when we want assigned a material property on the region.

#### 6.1.2. Geometrical attributes:

The attributes of primitive B-Rep geometry, namely those points, lines, circles are of common interest to the analyst. The Length of a line, distance of offset curve or surface, the radius of circle, and coordinates of points in space are typical attributes of this type of representation.
For example in STEP AP203 part 21, the offset surface is defined on Express language as follow:

```
ENTITY offset_surface
SUBTYPE OF (surface);
basis_surface  : surface;
distance       : length_measure;
self_intersect : LOGICAL;
END_ENTITY;
```

#### 6.1.3. Parametrical attributes:

- The form feature parameters like radius of through hole, ..
- Parametric description of geometric entities, like a number of control point of B-Spline curve,

#### 6.1.4. Physical attributes:

Boundary conditions: force, displacement.
Material property.

#### 6.1.5. Relationship attributes:

These apply to relation (dimensional, geometric and algebric constraints and tolerances of location ) between two pairs of entities in the shape model. For example parallel attribute may apply to a pair of faces, this type of attribute is important for idealisation process in order to characterise the idealisable area.

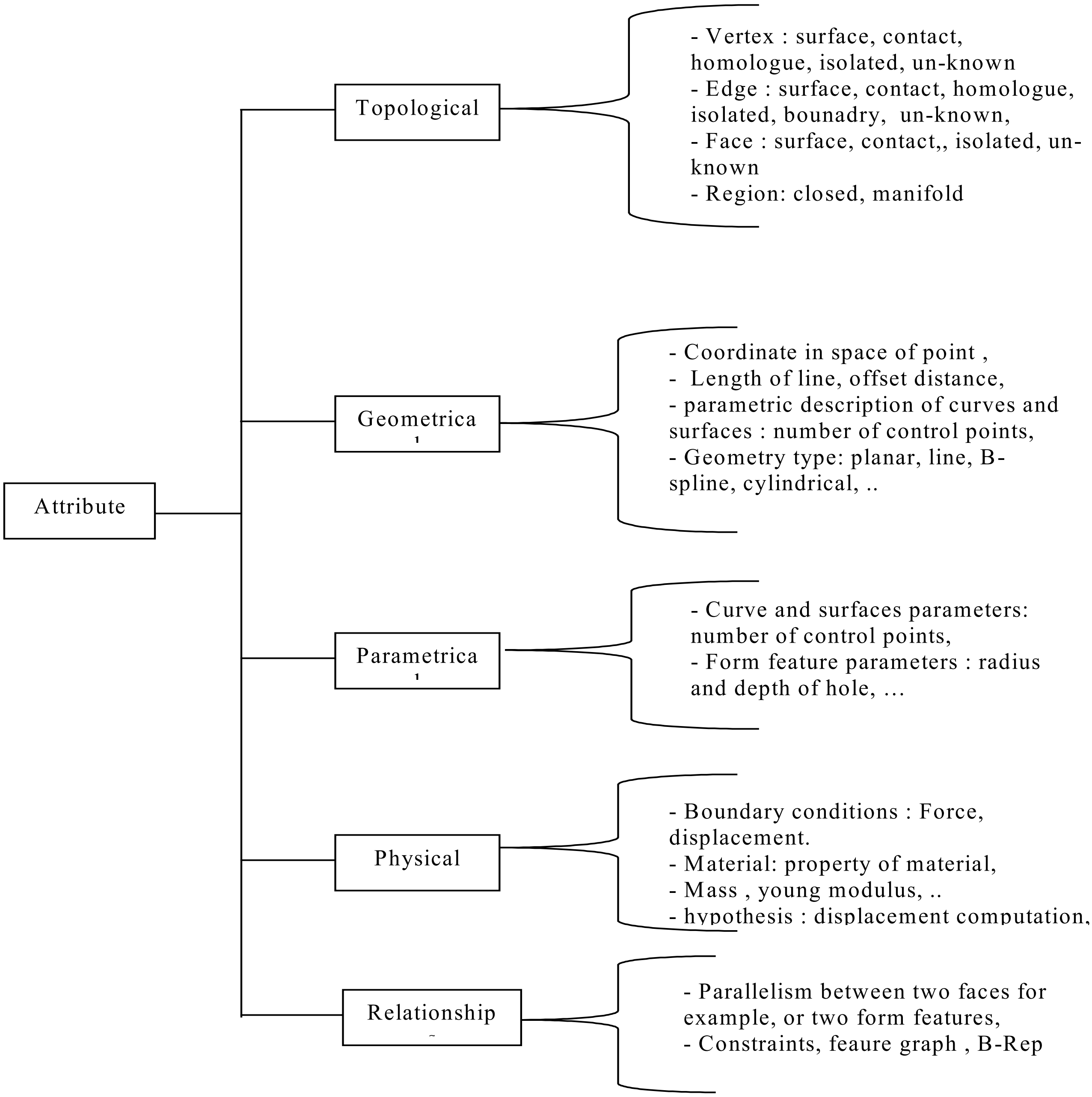



**Figure 33** Taxonomy of attributes

### 6.2. Attribute system:

The use of attributes has been addressed in the literature in general ways by Shephard and Beall [15-17], Peak an al [18-26] in F.E model context.
Shepherd considers the specification of analysis attributes from geometry based viewpoint as zero, one, and two dimensional attributes for respectively a point, curve, surface and volume. Three separate factors are considered for attribute specification, the dimension of the model entity, the dimensionality of the attribute and the distribution information which qualifies the attributes variation. Attributes are treated as tensor quantities which is a general approach for specifying and controlling them.
Peak and al propose an another approach based on the specific schema called APM (the analyzable product model) with allow to collect some information from the data bases of CAD systems via interfacing techniques (like geometric information from solid modeler, materials information from material data base,..) in order to create the idealized model, the proposed approach is dedicated to the idealization process which based on the same specific parametric relation in order to maintain the associativity between several sources of information.
The assoiatitvity process proposed by Peak an al [17-25] is based on tagging approach in order to extract the various idealizations attributes attached on the CAD model within CAD systems. The authors suppose that parameters attributes may be exported and imported in all three CAD systems (Pro/Engineer, IDEAS, and CATIA) , but an author parametric relation include for example the mass property as attribute, in this case CATIA does not support the reversible input/output.
The alternative solution is to separate the attribute system to the topological representation of attributes (HLT), for three main reasons:

- To allow the reuse of attributes without duplication the information,
- To check the validity of the HL Topology of attributes separately of the values of attributes during all process for analysis model generation,
- To keep the traceability of evolutions of the attributes during the simplification process, this point will contribute to compare the topology of models for the impact evaluations of shapes changes,

The proposed data structure is “reference key-driven” similar to the proposed data structure by OpenCascade(OCAF). The main deference between OCAF and the proposed attribute system is the hierarchy representation of attribute, when OCAF represent the attributes hierarchy as tree, our representation represent the attributes hierarchy as graph in order to taking into account the relationships between the attributes and also to maintain the Form feature graph along the process (see figure 34).
The reference key is implemented in the form of labels. Application data is attached to these labels as attributes. By means of the labels and the graph structure they are organized in, the reference key aggregates all the user data, not just shapes and their geometry. These are attributes like any other; no one attribute is master of the others.

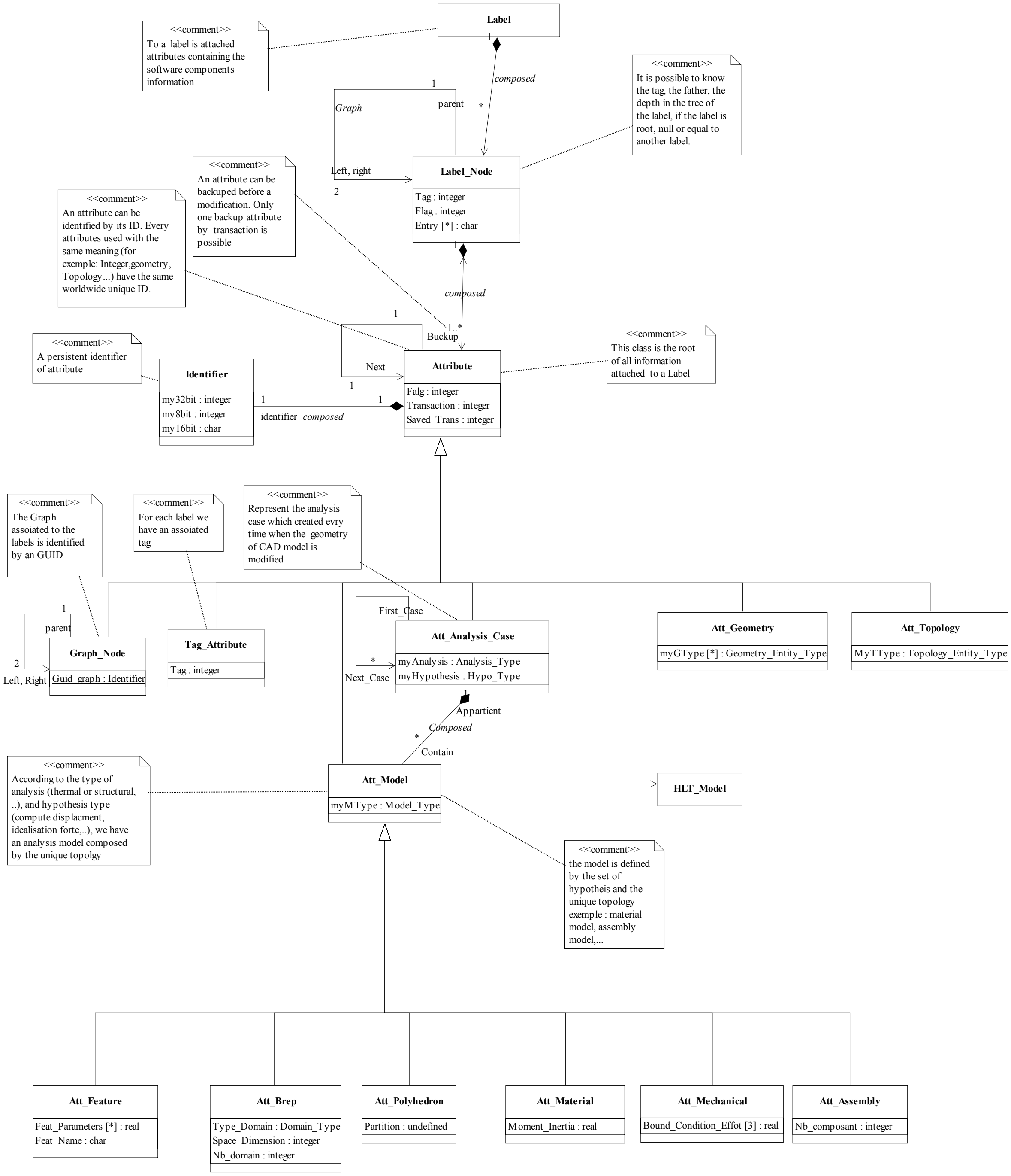



**Figure 34** Attribute system for analysis model preparation

The attribute system shown in the figure 34 is based on three concepts:

- The tag,
- The label,
- The attribute,

The first label in the system is the root of graph. Each label has a tag expressed as an integer value, and a label is uniquely defined by an entry expressed as a list of tags from the root, 0:1:2:1, for example (see figure 20).
Each label can have a list of attributes, which contain data, and several attributes can be attached to a label. Each attribute is identified by a unique persistent identifier (see identifier class in figure 32), and although a label may have several attributes attached to it, it must not have more than one attribute of a single identifier.
The sub-labels of a root label are called its children (left and right node of graph). Conversely, each label, which is not the root, has a father. Brother labels cannot share the same tag.
The most important property is that a label's entry is its persistent address in the attribute system.
A tag is an integer, which identifies in absolute identification; a label's place in the system is specified unambiguously by a colon-separated list of tags of all the labels from the one in question to the root of the attribute system.
The tag can be created in user-defined delivery; you assign it by passing the tag as an argument to a method.
The label itself contains no data. All data of any type whatsoever - application or non-application is contained in attributes. These are attached to labels, and there are different types for different types of data.

## *6.3. Organisational attributes*

Two basic concepts of organisational attributes which describe the complete information of an physical problem, this concept are represented as an graph shown in figure 34 and 35.

### 6.3.1. Analysis case:

The analysis case is the attributes represented as root of attributes graph, which defining the physical problem being modelled such type of analysis (structural, thermal, vibration, or composition: structural + thermal ...), and hypothesis type (displacement calculation, constraint calculation, idealisation forte, cisaillement,.), for each analysis case we have a set of Model_Entity represented as an graph (relationships between the HL Topologies). And for each analysis case is associated to the geometry of CAD model, when the geometry is modified a new analysis case is created for the modified geometry.

### 6.3.2. Model Entity:

According to the type of analysis we have a set of models entity (Material model, Mechanical model, Assembly model, Feature model, B-Rep model, and polyhedral model derived from Model_Entity) which is associated to the set of hypothesis, for each model entity is associated the set of attributes corresponding to the an unique topology (HLT).

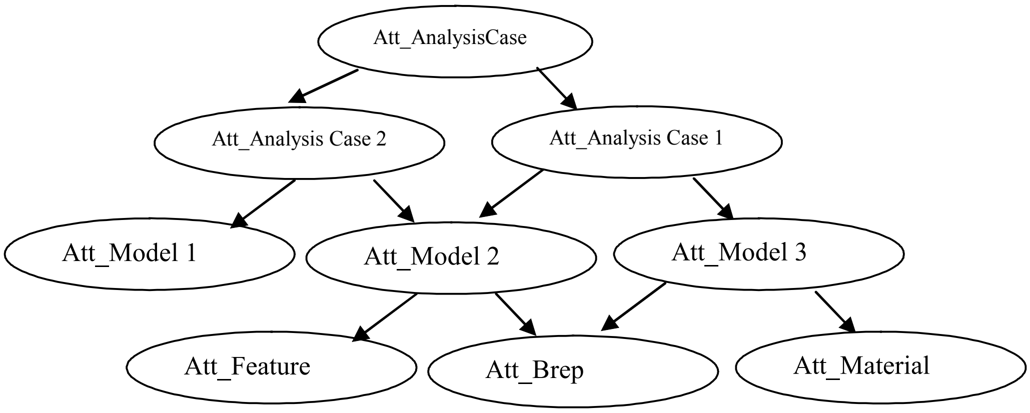



**Figure 35** Attributes of analysis case represented as graph of attributes

# 7. Attribute Propagation

## *7.1. Set up conformity*

*To be continued,*

## *7.2. Simplification process*

*To be continued,*

**The current work:**

1. Implementation of high level topology data structure: this data structure is done on the UML modeller but I think it is better to check its efficiency before the implementation.
2. Implementation of the partition data structure (schema 1): I already began the implementation of partition data structure based on polyhedral data structure (Simpoly).
3. Implementation of attribute system data structure: I now that the proposed system is not fully completed but for me the proposed system is independent for any software, and I think it's better to implement it.

4. Define the set of methods for the attributes (feature, face, boundary conditions., etc) propagation based on the attribute system during the set up conformity and the simplification process, it includes the face attribute inheritance mechanism (when a new triangle is added the mechanism to associate face attribute, see fig 36)

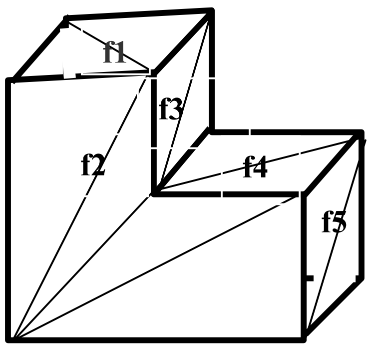



Figure 35: tessellted model

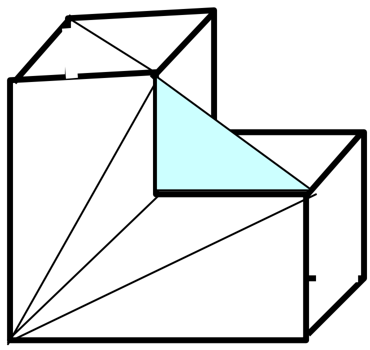


Figure 36: new triangle created after the simplification

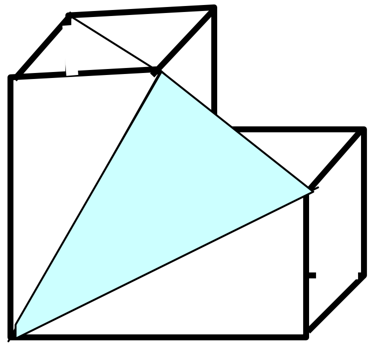


Figure37: new triangle created after the simplification

**Future work:**

5. Implementation (together with J-Philippe Pernot) of Feature-based Model,
6. Extension of Simpoly data structure to maintain features and face information
7. Define the method for the modification area localization, in order to compare two CAD models based on the attribute system.

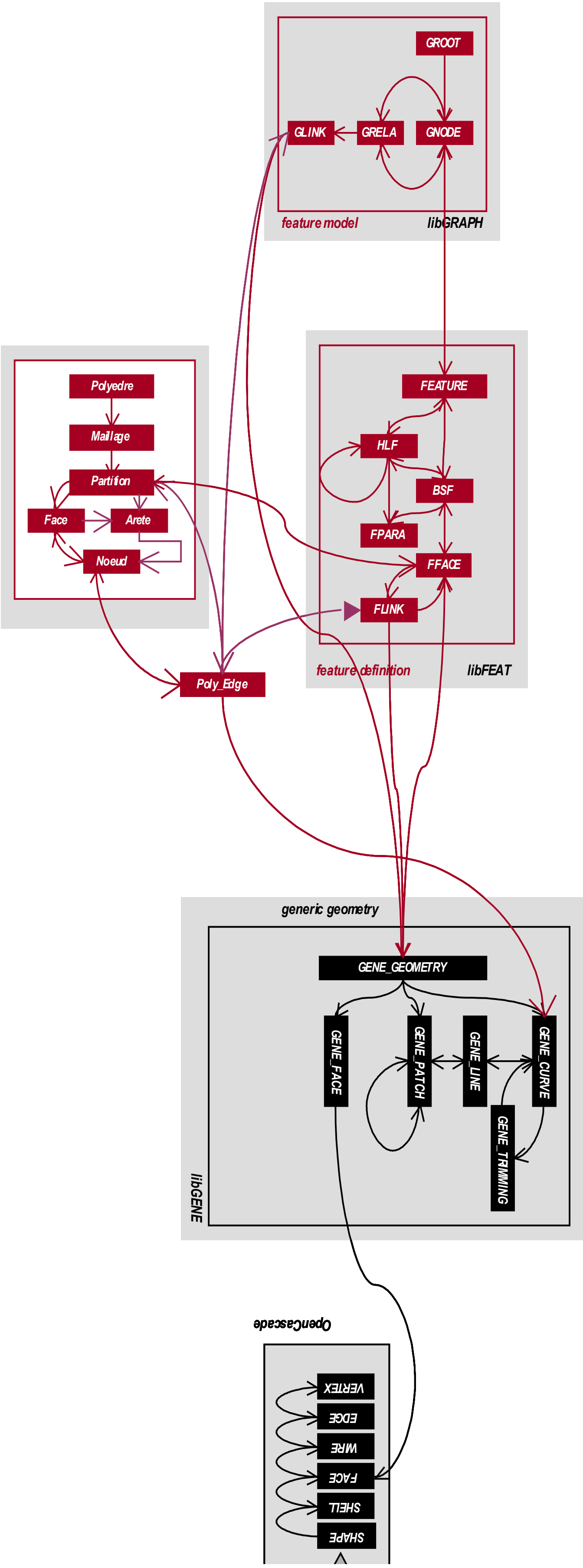



**Figure 36** schema 1 :

## 8. Bibliograpy:

[1] Dabke P., Prabhakar V., Sheppard S., (1994), "Using features to support Finite Element idealisation", International conference ASME, Minneapolis, Vol. 1, 183-193.

[2] Rezayat M., (1996), "Midsurface abstraction from 3D solid models: general theory and applications", CAD, Vol. 28, 905-915.

[3] Belaziz M., Bouras A., Brun J.M., (2000), "Morphological analysis for product design", CAD, Vol. 32, n°5-6, May, 377-388.

[4] Mobley V., Carroll M. P., Canann S. A., (1998), "An object oriented approach to geometry defeaturing for Finite Element Meshing", Proceedings of 7th International Meshing Roundtable, Sandia National Laboratories.

[5] Sheffer A., Blacker T., Clements J., Bercovier M., (2000), "Virtual topology operators for meshing", Int. J. of Computational Geometry & App., Vol. 10, n°3, 309-331.

[6] Fine L., Rémondini L., Léon J. -C., (2000), "Automated Generation of FEA models through idealization operators", Int. J. For Num. Meth. In Eng., Vol 49, n°1-2, 83-108.

[7] Fine L., (2001), "Processus et méthodes d'adaptation et d'idéalisation de modèles dédiés à l'analyse de structures mécaniques", PhD. thesis, INPG, Grenoble, France, July 6th.

[8] Véron P., Léon J. -C., (2001), "Using polyhedral models to automatically sketch idealized geometry for structural analysis", Engineering with computers, Vol 17, 373-385.

[9] Fine, L., Léon, J. -C., (2002), "A reverse engineering approach for FE model generation of historical constructions and industrial parts", 6th Int. Conf. On computational struct. Technology, Prag, September 4-6th.

[10] Shah J. J., Mantyla M., (1995), "Parametric and Feature Based CAD/Cam: Concepts, Techniques, and Applications", John Wiley & Sons, Inc.

[11] Giannini, F., De Martino, T., Falcidieno, B., (1997), "Context-dependent Recognition and Parametrization of Shape Features", IFIP WG.5.2 Workshop on Geometric Modelling in CAD, Airlie, Virginia, U.S.A., 18-23 May 19, Eds M.J. Pratt, R.D.Sriram, M.J.Wozny, Chapman & Hall.

[12] O'Bara R. M., Beall M. W., Shephard M. S., (2002), "Attribute management system for engineering analysis", Engineering with computers, Vol 18, 339-351.

[13] Panthaki M., Sahn R., Gerstle W., (1997), "An Object-Oriented Virtual Geometry Interface", Proc. 6th International Meshing Roundtable, Sandia National Lab., September.

[14] Fine, L., Rémondini, L., Léon, J. -C., (2000), "Topological operators to improve F.E.A. geometry adaptation", Fifth Int. Conf. on Computational Structures Technology, Leuven, September 6-8th.

[15] Ottmar Klaas, Mark S. Shephard: Embedding Reliable Numerical Analysis Capabilities into an Enterprise Wide Information System, Engineering with Computers 17 (2000) 2, 151 – 161.

[16] Beall,M.W.,Shephard,M.S.:An object oriented framework for reliable numerical simulations. Eng. Comp. (1999) 15, 61 - 72

[17] Beall,M.W. An Object-Oriented Framework for the Reliable Automated Solution of Problems in Mathematical Physics, PHD thesis.

[18] X-Analysis Integration Technology Russell S. Peak, Technical Report EL002-2000A

Engineering Information Systems Lab http://eislab.gatech.edu/ CALS Technology Center Georgia Institute of Technology Atlanta, Georgia USA, March 15, 2000.

[19] Tamburini, D.R.; Peak, R.S.; Fulton, R.E. (1997) The Analyzable Product Model: A Representation to Facilitate Design-Analysis Integration, AIAA Atlanta Section Aerospace Tech. Symposium, Atlanta.

[20] Peak, R.S.; Fulton, R.E.; Nishigaki, I.; Okamoto, N. (1998) Integrating Engineering Design and Analysis Using a Multi-Representation Approach, *Engineering wit Computers*, Volume 14, Number 2., 93-114.

[21] Koo, D.; Peak, R.S.; Fulton, R.E. (1999) An Object-Oriented Parser-based Finite Element Analysis Tool Interface, SPIE Intl. Symposium on Intelligent Systems and Advanced Mfg., Photonics East '99, Boston.

[22] Peak, R.S. (1999) Integrating Product Design and Analysis Models. An Overview of ProAM: Product Data-Driven Analysis in a Missile Supply Chain, 21st Century Commerce Expo, Assoc. for Enterprise Integration (AFEI), San Diego, CA, Track 7, Session 3.

[23] Peak, R.S. (2000) Automating Product Data-Driven Analysis Using Multifidelity Multidirectional Constrained Objects, NASA STEP for Aerospace Workshop, Jet Propulsion Lab, Pasadena CA.

[24] Peak, R. S. (2003), Characterizing Fine-Grained Associativity Gaps: A Preliminary Study of CAD-CAE Model Interoperability, 2003 Aerospace Product Data Exchange (APDE) Workshop, NIST, Gaithersburg, Maryland.

[25] R. S. Peak, R. E. Fulton, A. Chandrasekhar, S. Cimtalay, M. A. Hale, D. Koo, L. Ma, A. J. Scholand, D. R. Tamburini, M. W. Wilson (Feb. 2, 1999) Design-Analysis Associativity Technology for PSI, Phase I Report: Pilot Demonstration of STEP-based Stress Templates Georgia Tech Project E15-647, The Boeing Company Contract W309702.

[26] Zhou, W.X.; Fulton, R.E. (1995) Modularized and Parametric Finite Element Methodology for Concurrent Design, Analysis and Simulation, *Proceedings of Interpack '95 ASME International Electronic Packaging Conference*, Lahaina Hawaii.