Client : European Commission | Project : FRESH | Project N°: FP6-516059

Project Number:	FP6-516059
Document number:	3.2.2
Document Title:	Tools and algorithms able to convert files into PIVOT language
Document status:	Confidential

The aim of this report is to describe the different analysis and translation tools used for converting the information contained in AutoCAD and CATIA documents into PIVOT files.

Three tasks may be considered critical in this process. The first one is the analysis of the source files in order to extract the geometrical information of the mock-up that is needed for analyzing it. The second one consists in identifying the components and adding the electrical information. Finally, the third one is the translation of the structures resulting from the previous analysis to the correspondent PIVOT objects.

1.- AutoCAD files

Two different software tools called *AutoCAD to PIVOT Converter* and *PIVOT Editor* have been developed.

The first application, AutoCAD to PIVOT Converter, reads the information directly from AutoCAD files, identifies the geometric elements and generates a PIVOT file containing these data.

The PIVOT Editor is the application that offers the user a graphical view of the mock-up where he/she can identify the different elements of the mock-up and define their types. This operation would add the necessary electrical information and would also help to group the geometrical objects according to the elements of the mock-up they are part of. Once this definition process is completed the information would be saved in a new PIVOT file much richer and more elaborate than the previous one.

2.- CATIA files

A single application called Fresh3D has been developed that performs the different steps of the conversion process one after the other. However, Fresh3D does not read the information directly from the CATIA files, because these files are stored in an encrypted format. The solution chosen for solving this problem has been to export the CATIA mock-ups to a standard format, the STL, before reading and analyzing them.

Once we have the STL files, Fresh3D reads them and shows the mock-up in a 3D scene. In this scene, the user selects the different elements of the mock-ups and defines their types. This way a relationship is established between each geometrical object and the correspondent electrical device.

After this, the analysis of the mock-up may be performed. We can divide this process in four steps: a spatial partition of the mock-up with a classification of the triangles that compose the parts, the extraction of the connection points between the different elements, the calculation of the axis of the bundle segments and an analysis of the routes that extracts the paths that correspond to each wire.

- Spatial partition: this step is an improved version of the voxelization method. The voxelization is widely known and consists in a regular division of the scene in parallelepipeds or voxels (Voxel=unit of volume) of identical dimensions and the classification of the triangles that constitute the elements of the mock-up in the voxels that contain them. The voxelization is not the only method available. One of the alternatives to this spatial partition is to use an octree structure. The octree divides the bounding

box of the mock-up into eight octants, distributes the triangles of the parts in the correspondent one and then continues dividing recursively those that are partially full until the final resolution is reached. However, both methods have their drawbacks: the voxelization requires an excessive execution time for performing the analysis and the octree can not offer enough precision for a reasonable number of levels of division. In consequence, it was necessary to develop a hybrid analysis.
The hybrid analysis uses a spatial partition technique that combines both the octree structure and the voxels in order to achieve a fast analysis without losing precision. In order to improve the accuracy of the analysis, it classifies the leaf nodes of the octree in exterior, interior and border octants and, taking the last two groups, divides them in a voxelized mesh. Besides, this division of the leaf nodes is quite fast, because as the triangles of the parts have been pre-classified by the octree the voxelization does not have to handle so much data.

- Once we have obtained a mesh the next step is to calculate the connection points. The voxels that contain triangles of two or more parts are marked as candidates for calculating them and they are stored in lists, one for each couple of elements. When all the voxels have been checked, we take each one of these lists and calculate the medium point. These points will be the connecting points between the different parts. However, in complex mock-ups, the size of the voxels is often bigger than the distance between parts that are very close but that do not intersect and, as a result of this, false connection points are obtained. For solving this problem, it is checked (before storing the voxels in the lists) that at least a couple of triangles of both parts intersect. This increases the execution time, due to the additional test, but it gives excellent results in even the most complex cases.
- The third step of the analysis is the extraction of the axis of the bundle segments. This will allow us not only to obtain the routing of the wire but also to calculate its length. Firstly, the voxels that contain the bundle segment are classified in different layers. Secondly, a route is traced using the voxels of the most interior layers. Then, the voxel with the lower transition cost is chosen and the operation is repeated until the ending voxel is found. Finally, in the last step of the algorithm, the route is smoothed and stored in the correspondent structure.

  A pre-calculation of the neighbour voxels and octrees must be done before the extraction of routes, so that the transitions between voxels of different octants could be done correctly.

Initially, this was considered to be enough for obtaining the whole routing of a wire, but it was found that this only worked in those mock-ups where a single bundle connects directly the electrical devices. Unfortunately, in most cases, the bundles are divided in different segments that must be connected in order to put in contact the electrical devices, and, sometimes, there is an additional problem consisting in that the bundles may have branches that multiply the number of possible routes between the electrical devices. In consequence, a new analysis algorithm, called route analysis, was added.

The route analysis tries to calculate all the possible routes between two electrical devices. In a first step, one of the devices is chosen and a new route is created for each bundle segment connected to it. The process is repeated for each route. New sub-routes are added when more than one bundle, not already included in the route, is connected to the one currently being analyzed. Any possible loop is eliminated from the routes in a second step of the analysis. As this may produce routes that already existed, in the last step of the analysis, all the routes are checked for removing the duplicated ones.

Once the analysis of the mock-up has been performed and the electrical information has been obtained, the process of adding intelligence to the mock-up is complete. The user may then execute the function for generating the correspondent PIVOT files. This function will convert the structures with the results of the analysis to PIVOT objects and stores them in a file.