

# LAYOUT DRAWINGS OF THE LHC COLLIDER

A. Vergara Fernández\*, S. Chemli, B. Maire, Y. Muttoni, CERN, Geneva, Switzerland  
A. Kournossenko, R. Zalyalov, IHEP Protvino, Moscow, Russia

## *Abstract*

The team in charge of the LHC integration largely uses 3D scenes combining functional positions of equipments and the 3D CAD model issued from the CERN Drawing Directory (CDD) repository. This is made possible through the Digital Mock-Up tool developed at CERN. Giving dimensions in 3D context is a challenge with the current 3D CAD tools used at CERN. Requirements from users groups have made clear a need for automatic production of 2D layout drawings. This paper presents the retained solution to create on-request dimensioned drawings, to publish them, while maintaining coherence and consistency with the 3D integration scenes. Reliability of the information, on-line availability of the latest layout changes on dimensions and positions of equipments, and the maintenance of the facility will also be described.

## INTRODUCTION

The LHC integration team has been using largely the Digital Mock-Up tool (DMU) [1] in order to generate 3D renderings of the functional position of the machine equipments using the 3D CAD models provided by the 3D standards library under the CERN Drawing Directory (CDD) repository [2].

The installation of the machine inside the LHC tunnel has given rise to new requirements from the user groups (e.g. vacuum, integration) which have made clear a need for automatic production of 2D drawings in order to include new features. The new objectives are to use the DMU toolkit together with the CAD models to create dimensioned views of any area of the machine and add the possibility of getting top views of the equipments.

Unlike the mechanical layout drawings where the magnetic dimensioning was shown, these 2D layouts will not include the optical assimilation of the geometry.

## DRAWING CREATION

The 2D layout drawings are formed by several views whose number and arrangement depend on the kind of drawing that is requested (e.g. vacuum, racks, etc.). An Oracle table contains the number of views represented for each drawing type, the equipment that appear on each view, the dimensions and notes included and a set of parameters for the proper distribution of the views in the output.

Inside a drawing every view is created independently from the others. A view is formed by two main entities: first the geometry representing the top or the front view of

the equipment and, when required, the tunnel lines; and secondly the set of dimensions and notes properly computed and arranged dynamically in order to get an optimal display. When all the views included in the drawing are generated, an appropriate title block is created and displayed using data from the Oracle database and CDD.

## *Geometry*

The creation and retrieval of the geometry displayed for each view is completely independent from its linked dimensions. The DMU toolkit has been extended in order to add a new option for flattening the figures. The global coordinates are used for placing and flattening the geometry; afterwards it is translated to a local system coordinate shifted to the closest interaction point (IP). This is enough for providing the top and the front views of the last version of the machine equipment CAD models.

When required (mainly for the front views), the tunnel lines are also displayed. While the models of the equipments are retrieved from the repository and projected afterwards, the tunnel lines have to be computed every time the view is created. This is done by using the survey data of the measurements of the tunnel obtained from Oracle. These data have to be properly treated in order to distinguish singularities such as knot points from continuous tunnel slopes.

## *Dimension*

The core of the work carried out for creating this tool has been the automatic creation and display of the dimensions required by the users for each view. Dimensions are meant to be of two kinds: dimensions themselves and notes used for naming and displaying the exact location of the equipment.

The dimensions and notes are not done in the CAD system but in offline mode using the equipment data obtained from the LHC Reference Database [3]. When a view is created, the required dimensions and notes are not computed but just retrieved from the Oracle database and displayed together with the geometry. Of course, the retrieved data will not only be formed by the represented name or values, but they will provide information about the dimension type, its positioning within the view and other cosmetic parameters.

## *View Arrangement*

The user will not be allowed to personalize the requested drawing by choosing the number of views and the equipment and dimensions included. The different types of

\* Antonio.Vergara.Fernandez@cern.ch

drawings are defined in the database. An Oracle table defines the views for each drawing type together with their positioning (see figure 1). The different drawings types are being defined accordingly to the needs of the different user groups.

ID VIEW	VIEW TYPE	GEOMETRY	DIMENSIONS
30	FRONT	CRYOMAG DFB JACKS VACUUM MADREF	JACKS_DIM JACKS_NOTES CRYOASSY_DIM CRYOASSY_NOTE SFR NOTE SFR_DIM
31	TOP	CRYOMAG VACUUM MADREF DFB	VACUUM_NOTE VACUUM_DIM CRYOASSY_DIM
32	FRONT	CRYOGENICS DFB JACKS CARLOTHERQUE CRYOMAG RESERVED_SPACE	CARLOTHERQUE_NOTE RESERVEDSPACE_NOTE VOLRES_DIM
33	TOP	CRYOGENICS SFR VACUUM WEB CRYOMAG	CRYOLINE_NOTE CRYOLINE_DIM

DRAWING	ID VIEW	HOR_POS	VER_POS	SCALE
VACUUM	30	6	3	.02
VACUUM	31	6	5	.02
VACUUM	32	6	7	.02
VACUUM	33	6	9	.02

Figure 1: Example of tables for defining views within a drawing.

### Title Block and Sticker

The created drawings will be provided with a CDD number or a sticker depending on the type of drawing requested (official or not). The algorithm will get the updated CDD number and index version from an Oracle table and create the title block when required. The title block is completed with a list of changes from version to version. In the case of dealing with a non-official drawing (drawing on-the-fly) only a sticker is created indicating the associated CAD number.

## DIMENSIONING

### Creating and Dimension

All the dimensions needed for creating the LHC layouts are defined in two Oracle tables. The first one lists all the created dimensions and notes with the different parameters needed for their representation. This first table is enough for plotting a dimension but not for placing it into the view. The placement is done by using the second table, which is formed by the so-called attach points. On the first table each dimension has several attach points linked to it. The second table lists all the needed attach points with their related global coordinates (X, Y and Z). The number of attach points per dimension will depend on the dimension type (see Figure 2): a usual note, for instance, will require a single attach point, while a dimension will require two (begin and end) and some special ones like the L-Notes (see below) will require three.

The Oracle tables used by DMU provide the link of every equipment to its model in the CAD library together with a set of parameters to place it properly such as:

- **DcumStart**: cumulated distance from IP1 of the starting point.
- **DcumEnd**: cumulated distance from IP1 of the end point.
- **Local variables**: two extra parameters (U and V) needed for the anchorage of the assembly.

- **Anchor**: description of how to grip the assembly to place it and how to drop it in the accelerator using the local variables (e.g. beam axis, cryoline axis, tunnel axis, floor, etc.).

These parameters are also used to compute and position the dimensions. When a new dimension is created, the code computes the needed attach points. In order to keep uniqueness of the attach points, when an attach points is computed the program checks on the related table its existence and creates a new row only if it does not find it. This procedure minimises the length of the attach point table and simplifies the change management of a functional position. On the other hand, a dependence is created between dimensions that will have to be taken into account when data are removed or modified. Once the attach points are obtained, the rest of the parameters can be computed. For the notes the text to be written will depend on the type of equipment being named (e.g. position and name for the electronic racks, cumulated distance from IP1 and closest IP for the interconnection planes, etc). These data are directly obtained from Oracle. The lengths shown by the dimensions are computed when created using their attach points.

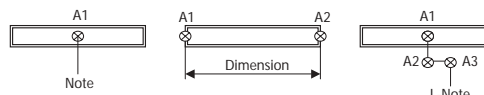


Figure 2: Number of attach points depending on the dimension or note created.

### Positioning and Cosmetics

In most of the created drawings the views will not only contain a single kind of equipment and, even for such simple cases, the dimensions and the notes will probably interfere and overlap each other. Since the display space on paper is scarce, each dimension will have to be placed taking into account the existence of others in order to get a view as clear and simple as possible. For this reason some of the cosmetic parameters will be defined when the dimension is created and stored in the Oracle tables but, by contrary, some of them will have to be computed when the dimension is displayed.

The most critical cosmetic parameters for properly arranging the dimensions and the notes is the distance between the attach points and the text: the offset. Some equipments are positioned in the tunnel within 20 cm, which gives on a 1:5 scaled drawing only some millimetres for displaying the geometry and the dimensions. The solution for avoiding overlaps has been treated differently for the dimensions than for the notes.

For the dimensions two kinds of overlaps can show up: overlap between equipments of the same family, which means that they will always appear since all the equipments will be retrieved when the family is called; and overlap between different dimensions for different families, for in-

stance when the cryoline and the cryomagnet lengths are displayed in the same view. The latter will be the most complex one since it will have to be solved differently for every view.

The overlap between dimensions of the same family is solved when the dimensions are created (see Figure 3). The algorithm in charge of creating the dimensions detects whether a problem can arise and sets an internal parameter for each dimension in order to specify the relative offsets between the dimensions within the family. This value will not be enough to place the dimension in the view since it will set which dimensions are closer and which are farther from the beam line but without giving the absolute value.

The interferences between different families are treated when the global picture is generated, hence the two dimension tables do not contain all the information about the placement. This allows a dynamic solution. A third table is defined in Oracle for storing the cosmetic details defined for each dimension type (e.g. kind of dimension, orientation, etc.). This table includes a parameter called offset level. This value defines the vertical order in which the different dimension families have to be represented in case that two or more are shown in the same view and at the same side of the beamline.

The problem of overlaps between notes (see Figure 4) has been solved in two different ways: exactly like for the dimensions or using the so-called L-Notes (see Figure 2). The user can define, when the drawing is created, whether to use one solution or the other and even a combination of them, allowing L-Notes to be applied to some families but not to others. All the note families for which L-Notes are allowed will be treated as a unique family. This method will avoid large offsets, which can be rather undesirable when several views are displayed in the same drawing.

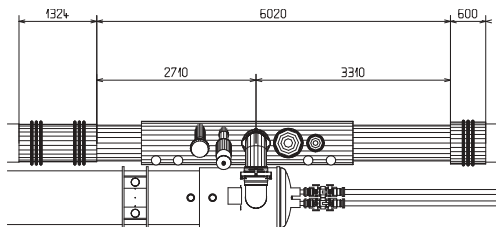


Figure 3: Automatic positioning of dimensions.

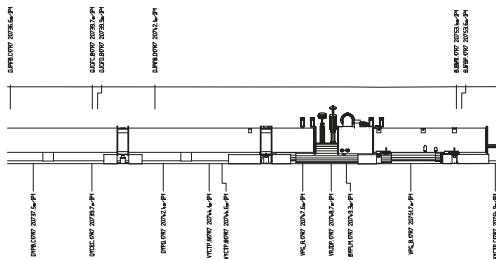


Figure 4: Automatic positioning of notes.

## GENERAL USAGE

The tool described so far has been conceived to be used for the generation of two kinds of drawings: official (created everytime that a new collider layout is released) and on-the-fly drawings (created at any time and not official).

The official 514 half-cells will be defined in an Oracle table that will be kept updated by bringing up to date the drawings concerned by the changes between two collider versions. Every official drawing will be identified by a CDD number and a version index which are provided to the Oracle table by EDMS. It will be possible to retrieve any of the versions of a drawing with a detailed list of the reason for change from version to version.

As said above, the on-the-fly drawings will not include a CDD number but a sticker and will be generated every time a user requests a certain kind of drawing for a LHC half-cell or half-cells. This procedure will be done via a Web service. The sole user of this tool will be the Integration Team, which will serve the requested drawings to the user groups.

## CONCLUSION

A tool for the automatic generation of dimensioned 2D layouts for the LHC machine has been developed in order to cope with the new needs from the different teams in charge of the collider installation. By summer 2004 this tool should be fully operational for providing official and on-the-fly drawings to the users.

## ACKNOWLEDGEMENTS

The authors would like to express their gratitude to Eric Van Uytvinck, Roger Valbuena and the rest of the IC-DC section: Pascal Le Roux and Antonio Jimeno-Yepes.

## REFERENCES

- [1] S. Chemli et al. "A Virtual CAD Model for the LHC", EPAC'00, Vienna, June 2000.
- [2] CERN Drawing Directory Web site URL <http://edms.cern.ch/cdd>
- [3] P. Le Roux, et al. "LHC Reference Database: Towards a Mechanical, Optical, and Electrical Layout Database", EPAC'04, Luzern, July 2004.