



**HAL**  
open science

## A hierarchical set of SysML Model-based objects for tolerance specification

D. Della Monica Ferreira, S. Patalano, S. Gerbino, Faïda Mhenni, Jean-Yves Choley

► **To cite this version:**

D. Della Monica Ferreira, S. Patalano, S. Gerbino, Faïda Mhenni, Jean-Yves Choley. A hierarchical set of SysML Model-based objects for tolerance specification. Systems Engineering (ISSE), 2016 IEEE International Symposium, Oct 2016, EDINBURGH, United Kingdom. pp.1-7, 10.1109/SysEng.2016.7753143 . hal-01589589

**HAL Id: hal-01589589**

**<https://hal.science/hal-01589589>**

Submitted on 6 Dec 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A hierarchical set of SysML Model-based objects for tolerance specification

F. Della Monica, S. Patalano  
Dept. of Industrial Engineering  
University of Naples Federico II  
Naples (NA), Italy  
fr.dellamonica@studenti.unina.it  
stanislao.patalano@unina.it

J.Y. Choley, F. Mhenni  
LISMMA laboratory  
Supméca – Institut Supérieur de Mécanique de Paris  
Saint Ouen Cedex, France  
{jean-yves.choley, faida.mhenni}@supmeca.fr

S. Gerbino  
DiBT Dept.  
University of Molise  
Termoli (CB), Italy  
salvatore.gerbino@unimol.it

**Abstract**— Modern engineering systems are getting complex and integrate multi-physical objects. The Model-Based System Engineering (MBSE) seems to be the best way to manage complex system design and the Systems Modeling Language (SysML) may be considered one of the computer languages to perform the designing of a complex system. MBSE also seems to be a valid solution to integrate tolerance specification into design process. In particular, in the present work SysML is used to create a set of libraries containing simple and complex volumes, primary datum and tolerance zones, according to ASME Y14.5M and ISO 1101 standards. The generation of these libraries is based on the Technologically and Topologically Related Surfaces (TTRS) model and uses the set of thirteen positioning constraints able to represent every condition between assembly features. The paper summarizes the characteristics of the created SysML objects, able to represent Datum, Datum Reference Frame (DRF) and tolerance zones. In particular, the Datum included in ASME Y14.5-M are modelled. Then, all the tolerance zones included in both standards are modelled. Finally, a three-step procedure is summarized to preliminary illustrate the way of use the developed set of SysML objects.

**Keywords**—*Model-based Systems Engineering; Assembly tolerance specification; TTRS model; SysML.*

## I. INTRODUCTION

Product design involves the assembling of parts or components, in which assembly tolerance specifications not only affects the quality of product, but also the manufacturing cost and service life of product [1]. During the design phase it is strategic to assure that dimensional and geometrical tolerance specifications are fully and correctly defined [2]. Defining assembly tolerance specification is the activity of specifying both assembly tolerance types and values able to assure assembly functional requirements [1]. Assembly tolerance specifications are usually carried out in conformance with tolerancing standards (e.g. ISO 1101 [3], ASME Y14.5-2009 [4]). These standards, therefore, do not provide a practical

approach to specify assembly tolerance types and values [5]. In particular, assembly tolerance specifications are specified by designers by using design drawings or within Computer Aided Design (CAD) systems. In this situation, assembly tolerance specifications largely depend on the judgment and experience regarding factors like functional requirements, selected materials and manufacturing processes. Different designers could specify different assembly tolerance specifications for the same nominal geometry. Therefore, during the designing of a complex product, the assembly tolerance specification will greatly increase the uncertainty in assembly performances and significantly affect the quality of the products.

Furthermore, the process of assembly tolerance specification for a complex system is a long and highly collaborative process. Designers have to synthetically consider geometrical features, functional requirements, tolerance types, tolerance values, and tolerance principles. This process often involves different designers operating at distributed sites and operating with heterogeneous CAD systems. There is a need for sharing the semantics of tolerance information among designers who do not necessarily share the same definition of terminologies, the same meaning of tolerance information, or the same background about an assembly tolerance specification design. So there is a critical requirement for process and information interoperability in tolerance specification activities for complex systems [1]. The Model-Based System Engineering (MBSE) approach seems to well respond to the above-mentioned needs.

Tolerance representation is significant together with tolerance specification as the way tolerances are represented often influences the way in which they can be specified and vice versa. Tolerance representation is mainly concerned with the following two problems [6], [7]. The first one deals with the organization and representation of tolerance information in a relative independent way, and therefore, it reflects the semantic distinctions of different types of tolerances. The other

one is to design a data structure and take it as carrier of storage and representation of tolerance information in computers. To well solve these problems, lots of tolerance representation models have been carried out. In Geometrical Dimensioning and Tolerancing (GD&T), a significant used structural model is the Technologically and Topologically Related Surfaces (TTRS) [8], [9].

This paper aims to provide a hierarchical set of objects to accomplish the tolerance specification by using SysML. In particular, the present work proposes the use of a Systems Modeling Language (SysML) and the use of TTRS model to create the objects to be composed, according to a general approach to tolerance specification.

## II. BACKGROUND

### A. TTRS Model

The functional dimensioning and tolerancing of a mechanical system represents a very long and difficult task, requiring skill and know-how. However, for several companies operating in automotive and aeronautics fields, it is most often a matter of redesigning rather than designing a new system or a new part. Therefore, it is of great relevance for companies to be able to declare a functional dimensioning and tolerancing regarding a class of mechanical parts that are independent from their geometric instantiation [9].

TTRS classification [10], [11] and, in particular, the set of relative positioning constraints, successively introduced, are useful to accomplish the dimensioning and tolerancing, allowing the declaration of parts in a way that is effectively independent from the respective geometric instantiation. TTRS model, in fact, provides that any surface could be classified in 7 different subsets or classes, according to their kinematic motion invariances. According to Figure 1 these classes are: spherical, planar, cylindrical, helical, revolute, prismatic, and complex. For each class, it is possible to associate the Minimal Reference Geometric Element (MRGE) that is the minimal combination of the following simpler geometric objects, named Reduced Geometric Element (RGE): point, line and plane [12].

Surface Classes	Degrees of freedom	Surface Classes	Degrees of freedom
1- Spherical surface	3 0 translation 3 rotations	5- Surface of revolution	1 0 translation 1 rotation
2- Planar surface	3 2 translations 1 rotation	6- Prismatic surface	1 1 translation 0 rotation
3- Cylindrical surface	2 1 translation 1 rotation	7- Complex surface	0 0 translation 0 rotation
4- Helical surface	1 1 combined translation and rotation		

Figure 1: Surface classes [13].

### B. Relative positioning modeling

The TTRS classification allows also the composition of different surfaces and their relative positioning by using 13 geometrical constraints, as depicted in Figure 2, between MRGEs [9]. It then allows the positioning and orientation of MRGEs.

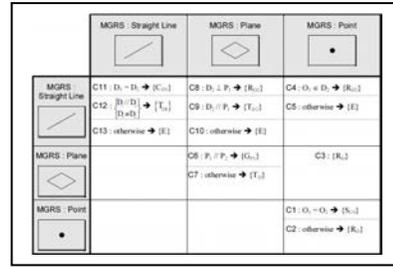


Figure 2: The thirteen relative positioning constraints [9].

TTRS allows also the creation of constraints between different parts of an assembly in order to take into account the kinematic joints between these parts, very useful for tolerancing. The set of constraints between MRGE of different parts being the same as TTRS theory one [12].

TTRS model is adapted to our case study because it includes the modeling of parts and their positioning through the consideration of their surfaces that facilitate its implementation in SysML extension. Another advantage of this model based on the Global Product Specifications (GPS) standards [14], [15], which are already implemented in CAD software, is that the user can formalize positioning constraints with the same “logic” he uses within a CAD tool. Finally, this approach allows to create all kind of geometry, simple or complex, so that the user can generate as complex geometries as he wants.

## III. SYSML GEOMETRICAL EXTENSION PROPOSAL

SysML is the language usually preferred by System architects for conceptual design. SysML is a general purpose modeling language for systems engineering. SysML is defined as an extension of a subset of the Unified Modeling Language (UML) using UML’s profile mechanism and was submitted to the Object Management Group (OMG) [16] in November 2005. A number of competing specification proposals were merged and adopted by the OMG as OMG SysML in July 2006. A profile in the UML provides a generic extension mechanism for customizing UML models for particular domains and platforms. Extension mechanism allows refining standard semantics in strictly additive manner, preventing them from contradicting standard semantics [17].

In order to define geometry some UML profiles or SysML extensions have been developed. For example, Böhnke et al. [18], who import in UML geometries issued from CATIA V5 design, but this method does not create directly geometry in UML, and, by using this profile, System architect is not able to specify any geometrical parameter specifications. Warniez et al. develop a geometrical SysML extension, that includes a library of simplified geometrical volumes, to define physical integration metrics, but this extension does not yet include TTRS constraints to manage relative positioning between components [18]. Barbedienne et al. have developed a SysML extension, to integrate geometrical paradigm in SysML modeling, named GERTRUDe (Geometrical Extension Related to TTRS Reference for a Unified Design) [12].

In order to take into account geometrical requirements and tolerance specification in the early stages of complex product design, starting from GERTRUDe, we propose a SysML extension for geometry and tolerances. This profile is aimed not only to easily specify geometrical requirements but also to model components of physical architectures, with a simplified geometry, and to specify related geometrical constraints and tolerances, based on the TTRS classification.

#### IV. GD&T MODELING IN SYSML

In order to accomplish the tolerance specification of a mechanical system the following set of objects were created: simple and complex volumes, primary datum, Datum Reference Frame (DRF), and tolerance zones.

##### A. Simple Volumes modeling in SysML

With reference to simple volumes, we have created an Assembly Block Definition Diagram (BDD) shown in Figure 3.

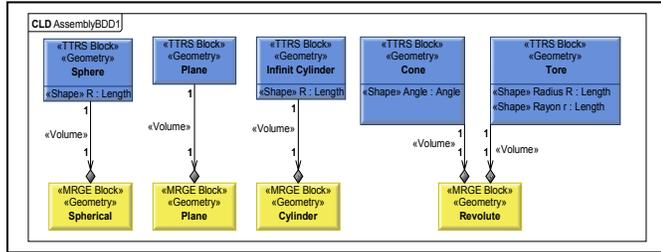


Figure 3: Simple volumes modeling in SysML

Figure 3 depicts the following simple volumes: “Sphere”, “Plane”, “Infinite Cylinder”, “Cone” and “Tore”. According to TTRS model “Sphere” is included in the class of Spherical surfaces, defined by the RGE “point” and a geometric parameter named “R=radius”. “Plane” is included in the class of Planar surfaces and it is defined by the RGE “plane”. “Infinite Cylinder” is included in the class of Cylindrical surfaces, defined by the RGE “line” and a geometric parameter named “R= radius”. “Cone” and “Tore” are included in the class of Revolute surfaces defined by RGEs “line and point”. “Cone” is defined by an angle and “Tore” by the following geometric parameters: Internal radius and External radius.

##### B. Complex Volumes modeling in SysML

By using simple volumes, the following complex volumes were created:

- Finite Cylinder;
- Finite Cone;
- Rectangle Parallelepiped;
- Opening Hollow Cylinder;
- Conical Hollow Cylinder;
- Hollow Cone;
- Hollow Rectangle Parallelepiped;
- Parallelepiped;

- Prism.

Due to the need of brevity, only the Assembly BDD related to “Finite Cylinder” is here summarized and depicted in Figure 4.

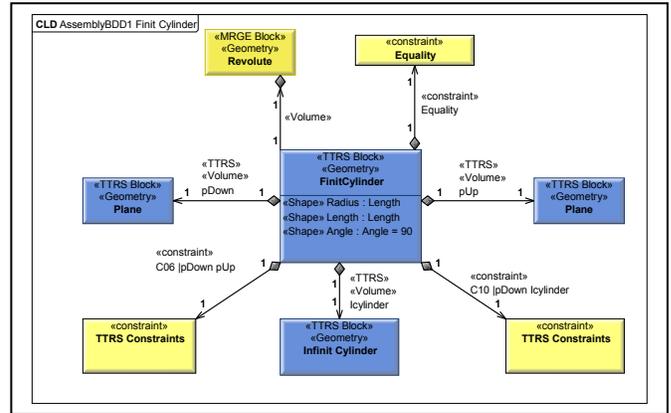


Figure 4: SysML modeling of “Finite Cylinder”.

This figure shows the “Finite Cylinder” made up of two “Plane” (“pDown” and “pUp” in Figure 4) and one “Infinite Cylinder”. “Finite Cylinder” is included in class of Revolute Surfaces. If we want to define a “Finite Cylinder” we have to specify its length and its radius. The positioning constraints applied are: C06 TTRS constraint defining parallelism and distance, representing length of “Finite Cylinder”, between the planes and C10 TTRS constraint defining perpendicularity between “Infinite Cylinder” and “pDown”. We have also applied a constraint of equality between radius of “Infinite Cylinder” and “Finite Cylinder” one. The management of constraints is modeled in an Assembly Parametric diagram shown in Figure 5.

The Assembly Parametric diagram shows interactions of different surfaces, and their corresponding positioning constraints between their related TTRS, MRGE, and RGE.

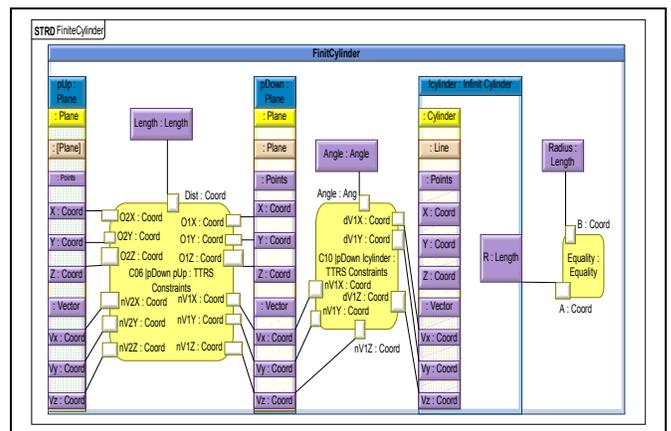


Figure 5: Assembly Parametric diagram of “Finite Cylinder”.

Some scripts have been developed, in order to facilitate user interface when using the profile. With these scripts it has been possible to add a geometric shape on a TTRS block.

When selecting a simplified geometry on menu, the script automatically generates corresponding geometrical dimension parameters (as *Block property values* of the TTRS block).

### C. Primary Datum implementation in SysML

In order to take into account tolerance specification, the creation of a library containing Primary Datum, with reference to Standard ASME Y14.5-M 2009 [4] was accomplished.

Therefore, with reference to Spherical and Cylindrical surfaces, two TTRS blocks, named “Datum: Spherical”, set with properties of “Sphere”, and “Datum: Cylindrical”, set with properties of “Finite Cylinder”, were introduced, respectively.

In order to define the Datum Feature Simulator (DFS), representing the theoretically perfect boundary used to establish a datum from a specified datum feature [4], a “Limited Plane”, which Assembly BDD is shown in Figure 6, was created by using the same procedure adopted for Simple and Complex volumes.

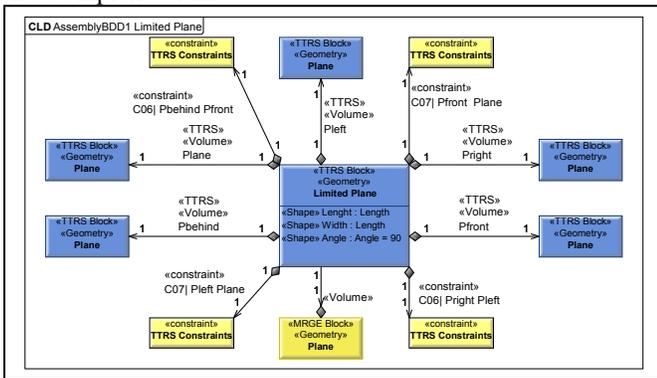


Figure 6: SysML modeling of “Limited Plane”.

Figure 6 shows “Limited Plane” made up of five “plane”, respectively, named: “Plane”, “Pright”, “Pleft”, “Pbehind” and “Pfront”. “Limited Plane” is included in class of planar surfaces. In order to define a “Limited Plane” its length and width have to be specified. The positioning constraints applied are: C07 TTRS constraint, specifying perpendicularity between “Plane” and “Pleft”, C07 TTRS constraint, specifying perpendicularity between “Plane” and “Pfront”, C06 TTRS constraint, specifying parallelism and distance, representing length of “Limited Plane”, between “Pleft” and “Pright”, C06 TTRS constraint, specifying parallelism and distance, representing width of “Limited Plane”, between “Pfront” and “Pbehind”. The management of constraints is modeled in an Assembly Parametric diagram shown in Figure 7.

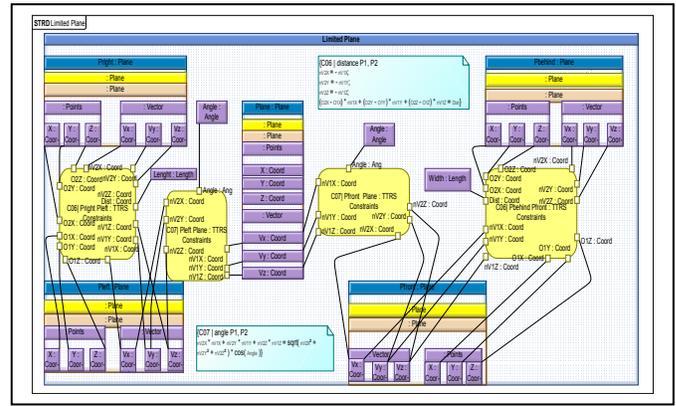


Figure 7: Assembly Parametric diagram of “Limited Plane”.

When the user selects “Limited Plane” on menu, an in-home developed script automatically creates the corresponding geometrical dimension parameters (as *Block property values* of the TTRS block).

With reference to Planar surface, a TTRS block, named “Datum: Planar”, set with properties of “Limited Plane”, was created. Then, by using “Limited Plane” it has been possible to define the Primary Datum “Width”.

The Assembly BDD related to Primary Datum “Width” is shown in Figure 8.

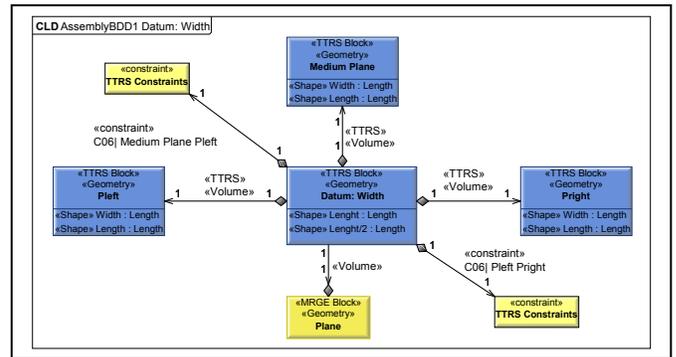


Figure 8: SysML modeling of “Datum: Width”.

Figure 8 shows the TTRS block, named “Datum: Width”, made up of three TTRS blocks, set with properties of “Limited Plane”, and respectively, named: “Pleft”, “Pright” and “Medium Plane”. “Datum: Width” is included in class of Planar surfaces. If we want to define the block “Datum: Width” is necessary to specify its length. The positioning constraints applied are: C06 TTRS constraint, specifying parallelism and distance, representing length of “Datum: Width”, between “Pleft” and “Pright”, C06 TTRS constraints, specifying parallelism and distance, representing length of “Datum: Width”/2, between “Medium Plane” and “Pleft”. The management of constraints is modeled in the Assembly Parametric diagram, shown in Figure 9.

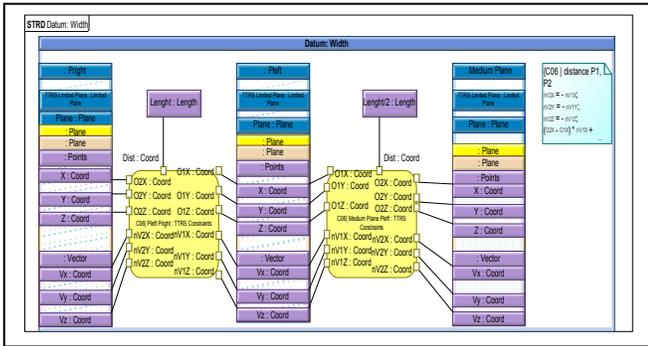


Figure 9: Assembly Parametric diagram of “Datum: Width”.

When the user selects a Primary datum on menu, the developed scripts create automatically the corresponding geometrical dimension parameters (as *Block property values* of the TTRS block).

#### D. Datum Reference Frame modeling in SysML

In order to accomplish tolerance specification, the following DRFs were created:

- DRF with mutually orthogonal planes (“DRF planes”);
- DRF with plane and axes (“DRF plane-axes”).

By using these TTRS blocks it is possible to specify a large set of tolerance zones, according to ISO 1101 or ASME Y14.5-2009 standards.

The Assembly BDD related to “DRF Planes” is shown in Figure 10.

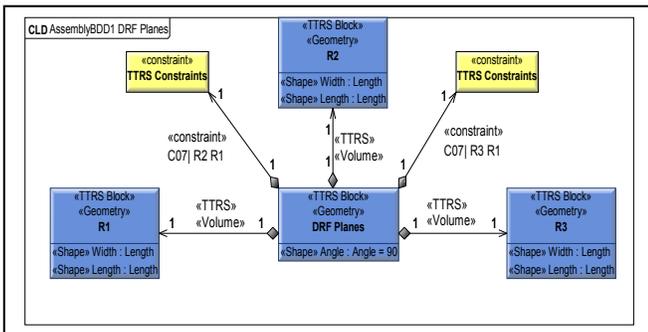


Figure 10: SysML modeling of “DRF Planes”.

Figure 10 shows the TTRS block “DRF Planes” made up of three TTRS blocks (“R1”, “R2” and “R3”) set with properties of “Datum: Planar”. The positioning constraints applied are: C07 TTRS constraint specifying perpendicularity between “R1” and “R2” and C07 TTRS constraint specifying perpendicularity between “R1” and “R3”. The management of constraints is modeled in the Assembly Parametric diagram shown in Figure 11.

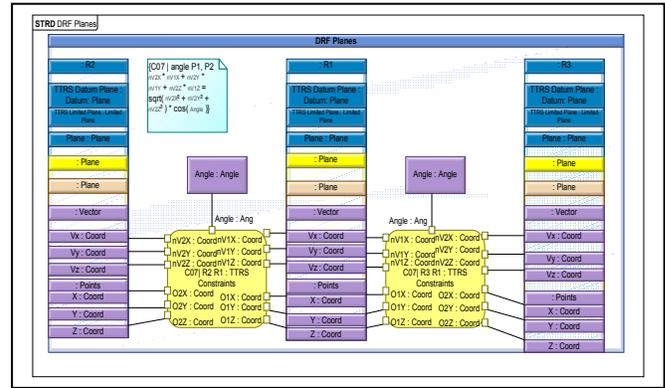


Figure 11: Assembly Parametric diagram of “DRF Planes”.

Further scripts have been developed, in order to facilitate the use of the profile. When user selects a DRF on menu, corresponding geometrical dimension parameters (as *Block property values* of the TTRS block) are created.

#### E. Tolerance zones implementation in SysML

By using the TTRS blocks previously created, the following tolerance zones were defined, according to UNI EN ISO 1101 standard [3]:

##### Tolerances of Form:

- Straightness;
- Flatness;
- Cilindricity.

##### Tolerances of Orientation:

- Angularity;
- Parallelism;
- Perpendicularity.

##### Tolerances of Location:

- Coaxiality;
- Symmetry;
- Position.

##### Tolerances of Run-Out:

- Total radial run-out;
- Total axial run-out.

Due to the need of brevity, only the Assembly BDD related to “Flatness” is here summarized and depicted in Figure 12.

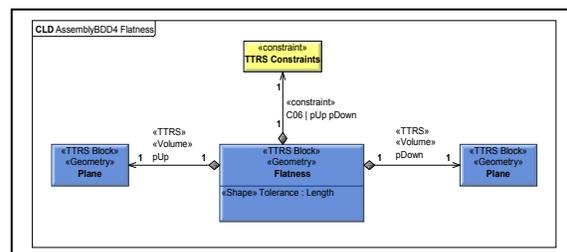


Figure 12: SysML modeling of “Flatness”.

Figure 12 shows the TTRS block, named “Flatness”, made up of two “plane” (“pDown” and “pUp” in Figure 12). Then, according to “Flatness” definition it is necessary to specify its tolerance value. The positioning constraint applied is C06 TTRS constraint specifying parallelism and distance, representing tolerance value of “Flatness”, between “pDown” and “pUp”. The management of constraint is modeled in the assembly parametric diagram shown in Figure 13.

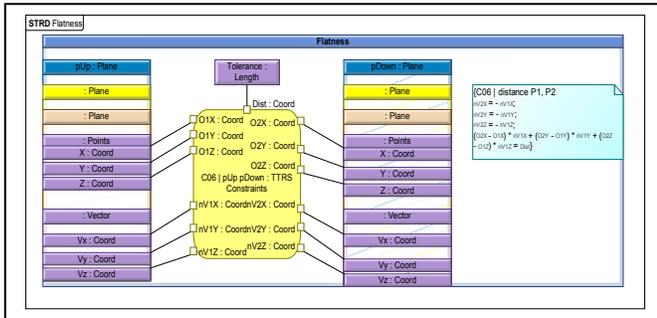


Figure 13: Assembly Parametric diagram of “Flatness”.

When the user selects a Tolerance on menu, a set of developed scripts automatically generates the corresponding geometrical dimension parameters (as *Block property values* of the TTRS block).

By using the developed simple and complex volumes, primary datum, datum reference frame and tolerance zones, it is possible to accomplish the GD&T specification of a complex system.

#### F. GD&T Specification

The procedure to accomplish the GD&T specification of a complex system uses the set of SysML objects created and it is characterized by three main steps. The first step deals with the creation, for each part of the assembly, of the TTRS blocks representing the features of the part. In particular, blocks are created for features to be toleranced and datum, thus allowing the definition of the Datum Reference Frame of the part (“DRF part”). Then, the second step deals with the creation, in an Assembly BDD, of a block for “Qualified Datum”, made up of the “DRF Part” and the objects belonging to the tolerance zones library while TTRS constraints between these objects are modelled in an assembly parametric diagram. This second step allows to accomplish the necessary datum qualification. Finally, in the third step, the blocks created in the previous steps are linked to the block representing the tolerances to be applied to the features of interest. The three-step procedure is then repeated for the successive part of the assembly, by taking into account the assembling sequence and the link between the feature of a part and the Datum of the successive part of the sequence.

## V. CONCLUSIONS

The present paper deals with a set of SysML objects aimed to integrate geometry, GD&T and constraints of a system during conceptual design phase within an MBSE approach. In

particular, the set of objects defined in the present work deals with simple and complex volumes, primary datum, Datum Reference Frame (DRF), and tolerance zones. Therefore, the defined set of objects could be used to preliminarily define a complete and unambiguous GD&T specification with reference to every physical component with canonical form, according to a MBSE approach. According to this aim, a preliminary three-step procedure was summarized in order to illustrate the way to use the created objects. The defined set of SysML objects is actually not useful for GD&T specifications related to components with non-canonical form. This restriction is due to the class of *Complex Surfaces*, defined by TTRS model, as such model actually does not provide a further classification of components with non-canonical form.

## ACKNOWLEDGEMENT

The present work was developed with the contribution of Campania Region-Italy performing the activities of the project LR. 5/2002-CUP E62I08000240002 carried out by University of Naples Federico II and Institut Supérieur de Mécanique de Paris – SUPMECA.

## REFERENCES

- [1] Y. Zhong, Y. Qin, M. Huang, W. Lu, W. Gao, and Y. Du, “Automatically generating assembly tolerance types with an ontology-based approach” *Computer-Aided Design* 2013, vol. 45, pp. 1253–1275, 2013.
- [2] P. Franciosa, S. Patalano, and A. Rivière, “3D tolerance specification: an approach for the analysis of the global consistency based on graphs” *International Journal on Interactive Design and Manufacturing (IJIDeM)* 2010, vol. 4, pp. 1–10, 2010.
- [3] ISO 1101 2004. Geometrical Product Specifications (GPS)—geometrical tolerancing—tolerances of form, orientation, location and run-out. Geneva, Switzerland: International Organization for Standardization; 2004.
- [4] ASME Y14.5-2009. Dimensioning and tolerancing. New York: American Society of Mechanical Engineers; 2009.
- [5] V. Srinivasan, “Standardizing the specification, verification, and exchange of product geometry: research, status and trends” *Computer-Aided Design* 2008, vol.40, pp. 738–749, 2008.
- [6] J. Guilford, and J. Turner, “Representational primitives for geometric tolerancing” *Computer-Aided Design* 1993, vol. 25, pp. 577–586, 1993.
- [7] M. Schulte, C. Weber, and R. Stark, “Functional features for design in mechanical engineering” *Computers in Industry* 1993, vol.23, pp. 15–24, 1993.
- [8] A. Desrochers, and A. Clément, “A dimensioning and tolerancing assistance model for CAD/CAM systems”, *The International Journal of Advanced Manufacturing Technology* 1994, vol. 9, pp.352–361, 1994.
- [9] A. Clément, A. Rivière, P. Serré, and C. Valade, “The TTRSs: 13 constraints for dimensioning and tolerancing” *Geometric Design Tolerancing: Theories, Standards and Applications*, pp 121-122.
- [10] A. Clément, A. Rivière, and M. Temmerman, “Cotation tridimensionnelle des systèmes mécaniques – Théorie et pratique”, *PYC Edition*, 1994.
- [11] A. Clément, A. Rivière, and P. Serré, “The TTRS: a Common Declarative Model for Relative Positioning, Tolerancing and Assembly”, *MICAD proceedings, Revue de CFAO et d’informatique graphique*, vol. 11, Hermès, pp. 149-164, 1996.
- [12] R. Barbedienne, O. Penas, J.Y. Choley, A. Rivière, A. Warniez, and F. Della Monica, “Introduction of Geometrical Constraints Modeling in SysML for Mechatronic Design”. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp 145-150, 2014.
- [13] D. Gaunet, “3D Functional Tolerancing & Annotation: CATIA tools for Geometrical Product Specification”, *Geometric Product Specification*

- and Verification: Integration of Functionality, Springer Netherlands, pp. 25-33, 2003.
- [14] ISO 17450-1:2011 - Geometrical product specifications (GPS) -- General concepts -- Part 1: Model for geometrical specification and verification ». [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_tc/catalogue\\_detail.htm?csnumber=53628](http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=53628). [Consulted on: 17-feb-2014].
- [15] « ISO 17450-2:2012 - Geometrical product specifications (GPS) -- General concepts -- Part 2: Basic tenets, specifications, operators, uncertainties and ambiguities ». [http://www.iso.org/iso/iso\\_catalogue/catalogue\\_tc/catalogue\\_detail.htm?csnumber=53629](http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=53629). [Consulted on: 18-feb-2014].
- [16] « OMG SysML ». <http://www.omgsysml.org/>.
- [17] S. Friedenthal, A. Moore and R. Steiner, "A Practical Guide to SysML" Second edition, Morgan Kaufmann publishers, 2012.
- [18] A. Warniez, O. Penas, R. Plateaux, R. Barbedienne "SysML Geometrical profile development for physical integration of mechatronic systems" IEEE/ASME International Conference on Advanced Intelligent Mechatronics 2014.