Science Arts & Métiers (SAM)
is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: https://sam.ensam.eu
Handle ID: .http://hdl.handle.net/10985/12537

To cite this version :

Elise GRUHIER, Frédéric DEMOLY, Samuel GOMES - A spatiotemporal information management framework for product design and assembly process planning reconciliation - Computers in Industry - Vol. 90, p.17-41 - 2017

Any correspondence concerning this service should be sent to the repository Administrator : scienceouverte@ensam.eu

A spatiotemporal information management framework for product design and assembly process planning reconciliation

Abstract

This paper introduces an innovative framework for product design and assembly process planning reconciliation. Nowadays, both product lifecycle phases are quasi concurrently performed in industry and this configuration has led to competitive gains in efficiency and flexibility by improving designers' awareness and product quality. Despite these efforts, some limitations/barriers are still encountered regarding the lack of dynamical representation, information consistency and information flow continuity. It is due to the inherent nature of the information created and managed in both phases and the lack of interoperability between the related information systems. Product design and assembly process planning phases actually generate heterogeneous information, since the first one describes all information related to "what to be delivered" and the latter rationalises all information with regards to "how to be". In other words, the integration of assembly planning issue in product design requires reconciliation means with appropriate relationships of the architectural product definition in space with its assembly sequence in terms of time. Therefore, the main objective is to provide a spatiotemporal information management framework based on a strong semantic and logical foundation in product lifecycle management (PLM) systems, increasing therefore actors' awareness, flexibility and efficiency with a better abstraction of the physical reality and appropriate information management procedures. A case study is presented to illustrate the relevance of the proposed framework and its hub-based implementation within PLM systems.

Keywords: Product Lifecycle Management, Ontology, Mereotopology, Assembly oriented design, Product architecture, Assembly planning, Reconciliation

Contents

1 Introduction **2**

2 Heterogeneity management issues in engineering **5**

3 Research background: Theory, ontology and approach **8**
3.1 JANUS as a spatiotemporal mereotopology-based theory . . . 8
3.2 PRONOIA2 as a spatiotemporal ontology 10
3.3 PROMA as an integrated management approach 12
3.4 Motivation 13

4 Spatiotemporal information management framework **14**
4.1 Management of product-process information consistency . . . 16
4.2 Information flow through PLM 20
4.3 Representation of design and assembly evolutions within a CAD application 31

5 Case study in a drones-based application **34**
5.1 Case study: Drones-based design for medical emergency . . . 34
5.2 MERCURY deployment regarding assembly evolution 36
5.3 MERCURY deployment regarding design evolution 45

6 Discussions **49**

7 Conclusions and future work **50**

1. Introduction

Over the last two decades, manufacturing industry has been progressively forced to compact its product lifecycles by ensuring a certain level of flexibility and efficiency as competitive edges, especially in the product development phase where design activities and decisions have a great impact on downstream processes. This has been done by considering the capture and integration of lifecycle constraints and knowledge in product design so as to deliver lifecycle friendly products. Nowadays, product design and assembly planning phases are almost concurrently processed in industry. This means that such effort has anyway provided competitive advantages in efficiency and flexibility by improving designers' awareness and product quality. Despite these efforts, some limitations are still encountered in product design regarding the lack of dynamical representation, information consistency and

Nomenclature	
PLM	Product Lifecycle Management
PDM	Product Data Management
MPM	Manufacturing Process Management
CAD	Computer-Aided Design
ASP	Assembly Sequence Planning
ASDA	Assembly Sequence Definition Algorithm
AOD	Assembly Oriented Design
JANUS	Joined AwareNess and Understanding in assembly-oriented deSign with mereotopology
PROMA	Product RelatiOnships Management Approach
PRONOIA2	PROduct relatioNships description based On mereotopologIcAl theory 2
MUVOA	MUlti-Views Oriented Assembly
MERCURY	a ManagEment framework appRoaCh of prodUct and process Relationships in assemblY and design phases
ASDA	Assembly Sequence Definition Algorithm
BOM	Bill Of Material
eBOM	engineering Bill Of Material
mBOM	manufacturing Bill Of Material
BOR	Bill Of Relation
DMU	Digital Mock-Up
GIS	Geographical Information System
OWL	Ontology Web Language
SWRL	Semantic Web Rule Language
XML	eXtensive Markup Language

information flow continuity. This is due to the inherent nature of the information created and managed in both phases and the lack of interoperability between the related information systems.

Product design and assembly planning phases actually generate heterogeneous information. Indeed product design addresses the description of all information aligned to "what to be delivered", while assembly process planning focuses on the definition of all information related to "how to be assembled". In other words, the integration of assembly planning issue in product design requires reconciliation means with appropriate relationships of the architectural product definition in space with its assembly sequence in terms of time. The entire understanding of how parts are assembled

within the embodiment design process is a critical issue and hence requires an appropriate context associated to assembly oriented design (AOD) issue. Actually the architectural definition of the product over the design process is defined by aggregating numerous design decisions/changes and multiple assembly constraints, which are generally triggered by either product architect or the designer, or better yet by the assembly planner in the AOD context.

From an information system point of view, product design and assembly process become more and more knowledge-intensive and therefore demand adapted and intelligent environment able to ensure dynamical representation, information consistency and information flow continuity in PLM systems. Currently, multiple PLM systems covers the beginning of the product lifecycle, such as product data management (PDM), manufacturing process management (MPM) and computer aided design (CAD) systems to name a few. Such systems have been introduced to ensure the management of the entire digital mock-up (DMU) of the product including its data structure and geometric definition, and its related manufacturing and assembly processes. As a result, some interoperability issues remain to be tackled since such systems consider spatial and temporal information without appropriate semantic and logical relationships [32]. Moreover, the fact of considering such heterogeneous points of view about the product development generally leads to inconsistencies, multiple design iterations and a lack of information continuity [49, 51]. This operational challenge is emphasised by considering the proactive nature of assembly oriented design philosophy, which leads to propose a specific orchestration of the product-process information flows including appropriate viewpoints.

Some similar issues exist in the domains of geographic information system (GIS) and building information modelling (BIM), where ontology model is commonly considered as a solution to solve heterogeneity issues [27]. As such, one promising way to overcome this limitation is to introduce semantic and logical layers to product architecture definition via a multi-layer ontology [12]. Built on this, PRONOIA2 (PROduct relatioNships description based On mereotopologIcAl theory 2) ontology has been developed and will be considered as a foundational basis for the proposed framework. The ontology will provide a useful bridge in order to connect current PDM, MPM and CAD data models and ensure information flow continuity and orchestration.

As a consequence, this paper focuses on product design and assembly planning reconciliation in the context of AOD, which intends to consider assembly process planning issue in the embodiment design stage in a proactive

manner. Here the main objective is to provide a spatiotemporal information management framework based on a strong semantic and logical foundation in PLM systems, increasing therefore actors' awareness, flexibility and efficiency with a better abstraction of the physical reality and appropriate information management procedures. In this context, the entire and seamless integration of assembly planning in product design has to be ensured with consistent relationships in the embodiment design phase to ensure a reliable product definition [29].

Firstly, the paper presents a brief literature survey on heterogeneity management issues in product design and assembly planning. Then, in section 3, research background introduces the JANUS (Joined AwareNess and Understanding in assembly-oriented deSign with mereotopology) theory based on mereotopology, the PRONOIA2 ontology, and the product relationships management approach (PROMA), considered here as the foundation of the proposed framework. Section 4 presents the core of the contribution with a spatiotemporal information management framework so as to manage product evolution from a design and assembly points of view. In section 5, a case study based on drones' design is described in order to illustrate the relevance of the proposed framework and its hub-based implementation within PLM systems. Finally, the advantages and limits of the framework are discussed in section 6, and conclusions and future work are given.

2. Heterogeneity management issues in engineering

Some research issues remain to be tackled in PLM systems and more especially in PDM and MPM systems [28], where critical information is still lacking in order to have a full understanding of the design and process activities. As an example, PDM systems currently manage purely spatial product information embedded in documents with various functionalities such as versioning, bill of material (BOM) management, workflow management, check-in/check-out procedures, change and configuration management and so on [21]. It intends to ensure that the right information is available for the right person at the right time and in the right format. Despite the information brought by PDM systems, few even none information about the relationships between parts and changes undergone by the product during its development is captured and managed. For instance, some comments on product documents' iterations enable the understanding of what has been changed between the different versions of a file, but a lack of semantic description of changes in such systems is highlighted. Besides, MPM systems

are restricted to manage temporal information. The description of relationships between assembly operations are actually limited to precedence and equal relationships [31].

The current issue of PDM and MPM systems in industry is their unidirectional way of reconciliation, which leads to inconsistent product definition over space and time, especially if a concurrent engineering strategy is adopted. Indeed, such procedure exists in MPM systems and enables a kind of continuity of product-process information via the engineering BOM (eBOM) and manufacturing BOM (mBOM) reconciliation link, but limited to a structural level. At this stage, it is vital to work at an architectural level, enabling the link of product and process models in the spatiotemporal dimension in order to better represent real phenomena, otherwise the lack of information about product evolution will be propagated on CAD models with a static description of the product. David and Rowe [6] went beyond such barrier and emphasised that PLM systems need a complex ontological model to process and integrate such information diversity. An effort to a single integrated system linking PLM systems in order to avoid heterogeneity issues has also been studied in [15].

Regarding information consistency in design engineering, Louhichi and Rivest [26] focused their efforts to maintain consistency between CAD work packages and the global product DMU . The same authors aim to control the DMU evolution, as well as the modification undergone. In addition, Chen and Luo [4] have made consistent the information travelling from design to construction phases. The benefits of such an approach are to get a better visualisation and coordination of architecture, engineering and construction with less mistakes and omissions, therefore improving the productivity and support for scheduling. Heterogeneity management is a recurrent issue, which is encountered in many engineering projects. For instance, BIM has attempted to manage spatial and temporal information. Spatial (i.e. building) and temporal (i.e. construction activities) information is supported during the development process of a construction project [24]. BIM also aids stakeholders to schedule activities and manage changes during the project [36]. A planning is generally placed on top of current design tools in the form of a time sequence. So the model is constructed with the following data structure: product, organisation and process. Contrary to the spatial and temporal information, here spatiotemporal dimension is not covered.

In the context of GIS, Straume [48] works with land management tools so as to have a consistent description over space and time. As such, land changes over time can be identified and team partners can collaborate. Moreover, Fonseca et al. [14] used an ontology to enable interoperabil-

ity, which is the ability of a system to share information with other system. Similar works, developed by Pittet et al. [35], try to make homogeneous the information exchange representation between stakeholders in the domain of facility management projects and BIM. An ontology has been built to organise and structure knowledge generated by each stakeholder during the lifecycle of the building. In cross-domains applications, CAD and GIS have been integrated by Peachavanish et al. [32]. This has facilitated interoperability between GIS (to perform location related to the analysis of components at different scales) and CAD (to capture and represent information about components) information. Such research works have provided better support in engineering and construction.

During the product development process, many versions of items and files are instantiated due to the different design options and alternatives generated. As such, each time a design change is made, the files are iterated or a new version is created. Gartner Group [37] states that 54% of manufacturing companies lack a unique repository to review and track changes across their products. As a consequence, specific procedures are needed to support changes and modifications so as to control information flows through the product lifecycle [46]. An engineering change is an alteration made to any parts, drawings or software that has already been released during the product design process [38]. The draft standard ISO 11442-6 presents the following examples [34], such as the change of a part depending on altered function or production requirements, the introduction of a new part or the replacement of a part. As such, the need to capture the change of position and geometry in space over time is highlighted [33]. Hoffman and Joan-Arinyo [22] define the events which imply changes (e.g. the element has been moved, deleted or joined with another element). Their classification focuses on elements, but not on the associations between these ones. Indeed, the product is continuously changing over its lifecycle and changes need to be captured, interpreted and understood. Making early design decisions has benefits but often requires modifications or engineering changes [44]. That is the reason why changes have to be managed in the early embodiment design stages in order to ensure a seamless control of product changes [10].

Engineering change management (ECM) is a business process in which product changes will be done (documents), and is part of configuration management in PDM systems [30]. Indeed, the fact of managing changes will increase stakeholders' knowledge on the product as they will better understand product and process evolutions [20]. ECM procedure controls changes workflow in order to keep track on who did what and when. The main research work in this domain focused on changes occurred during the man-

ufacturing and production phases. The impact of these changes on the product and on the delivered products is also studied [52]. In industry, ECM is recognised as an issue that receives too little attention relative to its importance [50]. Indeed, ECM becomes harder and harder when different data management systems (e.g. PDM and MPM) are implemented in the company [30]. Current PDM and MPM systems lack a full reconciliation enabling a consistent product definition and information traceability [46]. A solution is to create an environment that encourages a strong relationship between the engineering and manufacturing departments [45].

Figure 1 maps existing management frameworks from literature according to the type and the dimension of the entities to be managed, as well as the temporal position in the product life cycle. Such representation highlights a lack of spatiotemporal information (i.e. changes) management since most of the frameworks focus on the management of information in the spatial dimension and few of them in the temporal dimension. As a consequence, this figure introduces the MERCURY framework to manage spatial, temporal and spatiotemporal information in the context of AOD so as to support product design and assembly planning reconciliation. This framework will be presented in more details in later sections of this paper.

3. Research background: Theory, ontology and approach

This section aims to introduce the foundation on which the proposed framework is based. It actually describes a theory built on a perdurantist philosophy and a dedicated multi-layer ontology (i.e. meta-, domain- and application-ontology) [**?**].

3.1. JANUS as a spatiotemporal mereotopology-based theory

A mereotopological theory, called JANUS, has been introduced in [19], in order to formalise tacit knowledge covering product design and assembly process planning phases. The main objective of JANUS is therefore to qualitatively describe product evolution (i.e. in terms of changes over time) in the context of AOD based on a perdurantism philosophy. Perdurantism states that an object can evolve over time and space, since it is composed of different temporal parts through its existence. This theory is based on Smith's spatial mereotopological primitives [42] and has been extended to the temporal and spatiotemporal dimensions. Smith's primitives (such as ***Part of***, ***Overlap*** or ***Tangent***, etc.) are seen as useful to describe the physical relationships between two mechanical parts and are shown in Figure 2 and Table 1. The temporal dimension is inspired from Allen's proposal

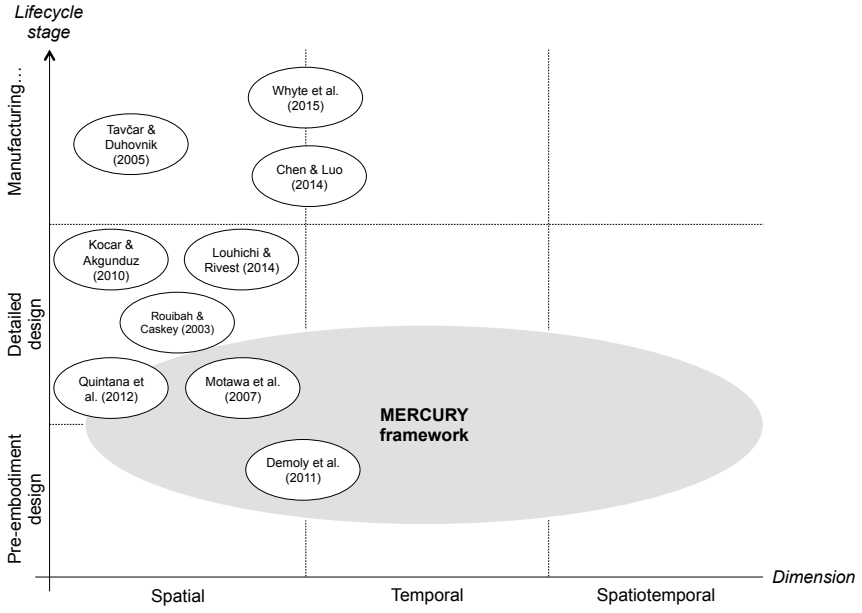

Figure 1: Map of existing management frameworks over lifecycle phases and dimensions

[1] and the previously developed spatial entities. So temporal primitives can associate two assembly operations and are presented in Table 1.

In addition to the spatial and temporal dimensions, a novel dimension has been developed so as to describe objects evolution over space and time. The spatiotemporal dimension highlights two novel notions, called spatiotemporal regions (i.e. swept volume) and spatiotemporal primitives (i.e. classified into design changes, kinematic and technological pairs positioning) [41, 47]. Here, design changes represent the intrinsic evolution of a product over time and kinematic pair positioning defines how a part is moving to be positioned in a specific manner with its physically connected parts. Mereotopological primitives have also been developed to enable the capture of design intents through the AOD process [40, 39]. Based on such a theory, the product evolution in the context of AOD can be described and fully understood by the product architect. Moreover, the JANUS theory has been built to also dynamically represent the product from design and assembly planning perspectives. As such, spatial (i.e. mechanical part) and temporal (i.e. assembly operation) objects can be related in the early design

stages via appropriate spatiotemporal relationships.

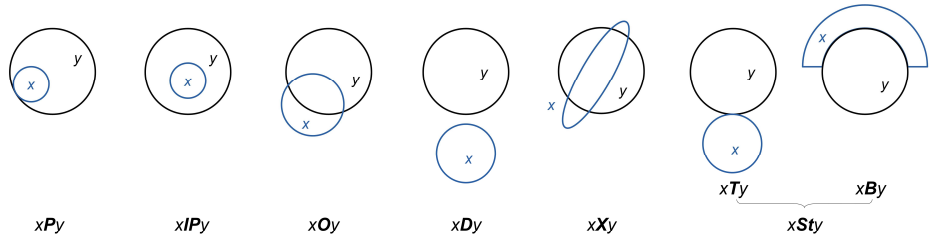

Figure 2: Smith's spatial mereotopological primitives [42]

3.2. PRONOIA2 as a spatiotemporal ontology

Based on JANUS mereotopological theory, PRONOIA2 ontology has been developed in order to be machine interpretable within PLM systems [18]. The objective is to solve the heterogeneity issue with semantics and logics so as to establish relationships among different dimensions and domains, as well as to ensure the management of information from PDM and MPM and CAD systems from a central place. PRONOIA2 is composed of a meta-, domain- and application-ontology. Figure 3 presents a part of the ontological model. It has been developed using Protégé Editor and Ontology Web Language 2 (OWL2) to describe classes and properties. Based on this knowledge structure, description logics (DL) and semantic web rule language (SWRL) rules have been set up with restrictions in order to facilitate semantic reasoning and consistency checking. Here information consistency can be checked during the design process with HermiT reasoner in Protégé. In addition, PRONOIA2 enables stakeholders (i.e. product architect, assembly planner and designer especially) to work collaboratively using the same semantics and makes the information exchange more effective [16]. This ontology aims to be integrated with PDM, MPM and CAD systems so as to ensure better information flows and then better data utilisation for decision making in AOD. As a result, existing heterogeneity gap can be minimised, since spatial and temporal information can now be reconciliated from a semantic and logical point of view. Last but not least, the ontology is expected to provide a better understanding [43] of the evolution of the product structure in the context of AOD.

Table 1: Spatial and temporal mereotopological primitives used in JANUS [19]

	Definition	Mereotopological description
Spatial	x is part of y	$x\mathbf{P}y := \forall z(z\mathbf{O}x \rightarrow z\mathbf{P}y)$
	x overlaps y	$x\mathbf{O}y := \exists z(z\mathbf{P}x \wedge z\mathbf{P}y)$
	x is discrete from y	$x\mathbf{D}y := \neg x\mathbf{O}y$
	x is InteriorPart of y	$x\mathbf{IP}y := x\mathbf{P}y \wedge \neg x\mathbf{T}y$
	x crosses y	$x\mathbf{X}y := \neg x\mathbf{P}y \wedge \neg x\mathbf{D}y$
	x straddles y	$x\mathbf{St}y := \forall z(x\mathbf{IP}z \rightarrow z\mathbf{X}y)$
	x is boundary of y	$x\mathbf{B}y := \forall z(x\mathbf{P}x \rightarrow z\mathbf{St}y)$
	x is tangent of y	$x\mathbf{T}y := \exists z(z\mathbf{P}x \wedge z\mathbf{B}y)$
Temporal	a is temporally part of b	$b\mathbf{Pt}a := \forall c(c\mathbf{Ot}b \rightarrow c\mathbf{Pt}a)$
	a is start temporally part of b	$b\mathbf{Pts}a := \exists c(b\mathbf{Pt}a \wedge c\mathbf{Tt}b \wedge c{<}a)$
	a is finish temporally part of b	$b\mathbf{Ptf}a := \exists c(b\mathbf{Pt}a \wedge b\mathbf{Tt}c \wedge a{<}c)$
	a is temporally interior part of b	$b\mathbf{IPt}a := b\mathbf{Pt}a \wedge \neg b\mathbf{Pts}a \vee \neg b\mathbf{Ptf}a$
	a temporally overlaps b	$b\mathbf{Ot}a := \exists c(c\mathbf{Pt}b \wedge c\mathbf{Pt}a)$
	a temporally precedes b	$a{<}b := \exists c(a\mathbf{Tt}c \wedge b\mathbf{Ptf}c)$
	a is temporally tangent to b	$b\mathbf{Tt}a := \exists c(c\mathbf{Pts}a \wedge c\mathbf{Bt}b)$
	a is temporally equal to b	$b{=}a := b\mathbf{IPt}a \wedge a\mathbf{IPt}b$

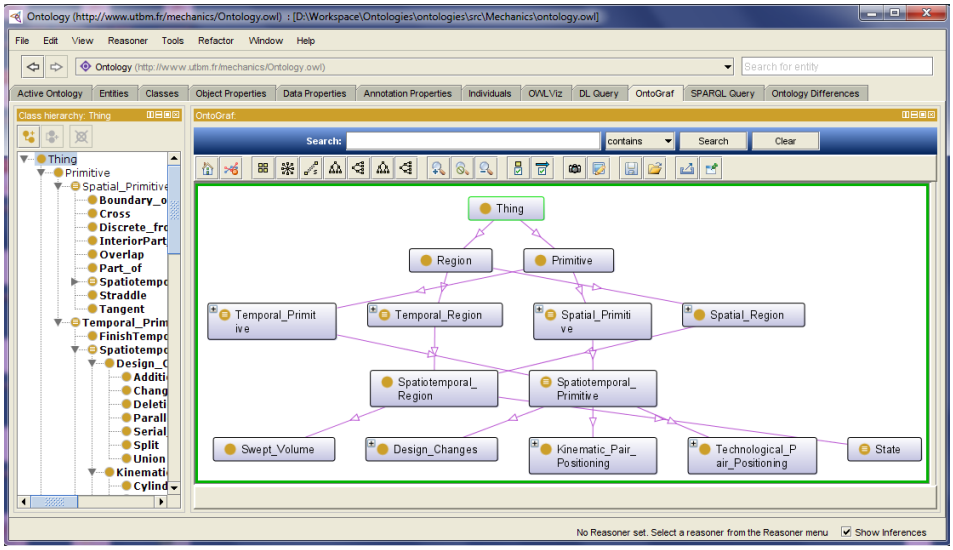

Figure 3: Top-level vision of the PRONOIA2 ontology classes

3.3. PROMA as an integrated management approach

PROMA (PrOduct Relationships Management Approach) has been developed to cover product relationships management issue in PLM [7] so as to increase their capabilities. In this approach, product-process information is reconciliated via a bill of relations (BOR), where part-to-part relationships are captured but not defined according to a spatiotemporal semantic. Such relationships can be related to assembly skeletons that are managed in a CAD BOM so as to support designers' activities in CAD modelling stage. At this stage, no interface skeleton, such as introduced in JANUS theory, has been considered yet. Moreover, a MUVOA (MUlti-Views Oriented Assembly) model [8] has been presented to highlight the link between product architect's and assembly planner's viewpoints in their related domains (i.e. respectively product design and assembly planning). It can be observed that an interface domain is not specified in this MUVOA model because of the lack of spatiotemporal information integration. The PROMA approach will be considered as a strong basis for the elaboration of the MERCURY framework in order to cover towards the spatiotemporal dimension, which enables considering product evolution in the context of AOD and changes in product design. By adding such a critical dimension, it will be needed and possible to (i) manage spatiotemporal information, (ii) keep traceability of

product-process information and (iii) reason at a logical and semantic level in order to ensure information consistency in product design and assembly planning phases.

3.4. Motivation

As stated in the above-mentioned sections, reconciliation between product design (covered by PDM systems and CAD applications) and assembly process planning (covered by MPM systems) information is widely considered as a huge challenging issue. Indeed, such heterogeneous information needs to be managed with a seamless integration so as to cover spatial, temporal and spatiotemporal issues. By considering spatiotemporal description in PLM systems, the product evolution in terms of design changes and assembly will be captured and managed so as to increase awareness of involved actors in the AOD context. Built on this, needs have been highlighted to develop a spatiotemporal information management framework supporting product design and assembly planning reconciliation and enabling a consistent product description over space and time based on a previous elaborated theory and developed ontology (Step 3 of Figure 4). Such ambition represents the last part and the outcome of a much larger research project, which aims to support the description and representation of object-process operation relationships over time and space within PLM systems.

Built on these stakes, Figure 5 introduces the integration of a novel framework called MERCURY (ManagEment framework appRoaCh of prodUct and process Relationships in assemblY and design phases) within previously developed PRONOIA2 ontology and JANUS theory. The objective of the theory is to qualitatively describe product-process relationships so as to support product evolution in the context of AOD [19]. Such mereotopological formalisation has enabled the development of a multi-layer ontology in order to tackle heterogeneous issues in assembly oriented design [18]. Here the theory and the related ontology provides a semantic and logical foundation to associate spatial and temporal information so as to reconciliate PDM and MPM systems in an efficient manner. By bridging these two information systems, product design and evolution over time can be managed so as to enhance dynamic aspect in product design. The main objective of MERCURY is to introduce a novel information management layer (i.e. spatiotemporal layer) in the design process in order to manage assembly and design changes that a product undergoes. With such a framework, a better articulation between product design and assembly planning is enabled. As a result, the MERCURY framework aims to (i) reconciliate heterogeneous information in the spatial and temporal dimensions in the beginning of the

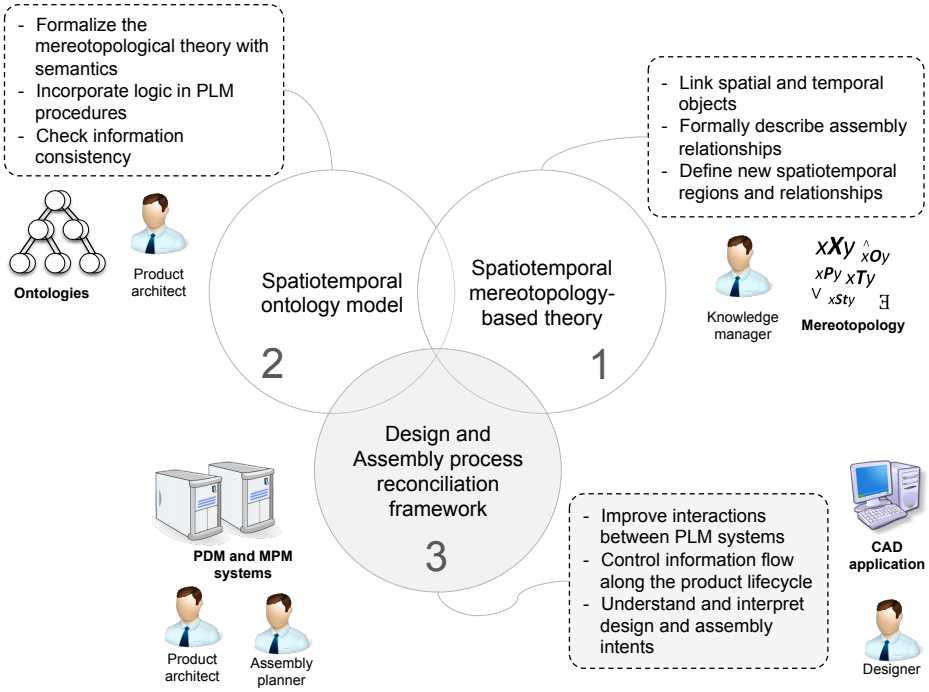

Figure 4: Overall vision of the research

embodiment design stage, (ii) manage and control the reconciliation from PDM and MPM systems and (iii) integrate the reconciliated information within CAD applications.

4. Spatiotemporal information management framework

The proposed spatiotemporal information management framework (MERCURY) can be applied on two different applications: (i) the management of assembly evolution in the context of AOD and (ii) the management of changes during the design process. In the first application, assembly evolution mainly concerns the product move during the assembly process (e.g. kinematic and technological pairs positioning). On the other hand, the second application is focused on design evolution by describing design changes during the design process.

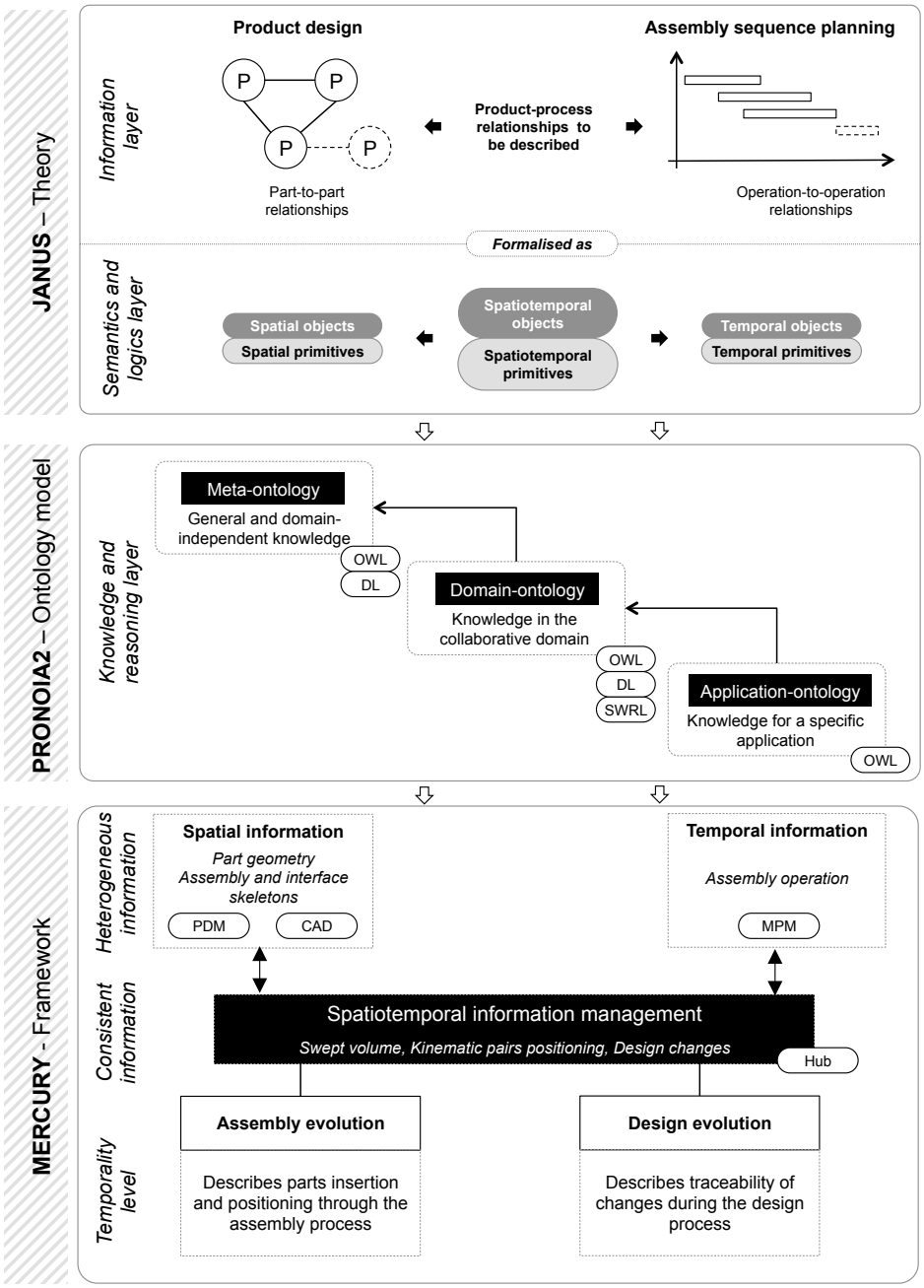

Figure 5: MERCURY framework based on PRONOIA2 ontology and JANUS theory

4.1. Management of product-process information consistency

In the context of AOD, two dimensions can be considered since product design relates to the spatial (e.g. mechanical part) dimension and assembly process planning to the temporal (e.g. assembly operation) one. Indeed in current engineering activities, the product architect mainly deals with spatial objects while the assembly planner works with temporal objects. Here the main objective is to associate spatial and temporal objects via spatiotemporal objects (i.e. swept volumes) and relationships such as previously described in JANUS theory (see Figure 6). The management of such spatiotemporal relationships will ensure a better interaction, coordination of product-process information [3] and information flow consistency. This will lead to the consideration of changes and a consistent product definition over space and time.

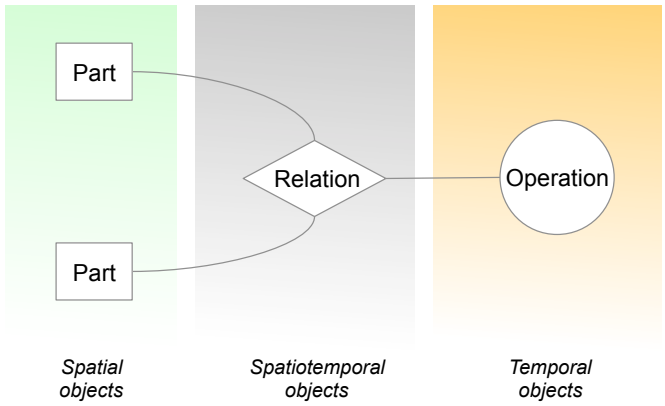

Figure 6: Association between spatial and temporal objects through spatiotemporal relationships

From another perspective, the fact of associating spatial and temporal objects is seen as a critical step towards the reconciliation of data structures in PDM-MPM systems and CAD application. Figure 7 shows the introduction of the bill of relations (BOR) in order to capture and manage part-to-part relationships over time from a central place. This means that the set of information flows can be orchestrated from this specific structure, and all related structures (i.e. eBOM, CADBOM, mBOM and BOP) can be managed in a consistent and bidirectional manner. PDM systems usually manage data structures within the eBOM and related product configurations, while MPM systems work with mBOM and BOP (bill of processes)

to define assembly sequences and process plans. The idea is then to incorporate spatiotemporal relationships within a BOR in the embodiment design stage, so as to bridge the gap of information heterogeneity between product design and assembly process planning. Here, spatiotemporal relationships are defined as novel technical objects, which link part-to-part relationships (including spatial primitives and spatial objects) with assembly operations (temporal objects) so as to generate swept volumes (spatiotemporal objects). Spatiotemporal relationship is vital to describe part-to-part relationships over time and also design changes during the product design process. This novel primitive (e.g. ***RevoluteOP***) can be directly associated to a part-to-part relation (e.g. revolute pair) within an assembly oriented context. Actually, the revolute pair only describes the final state of the kinematic pair between two physical parts, while ***RevoluteOP*** represents all states when two discrete parts are moving to be positioned in a revolute manner.

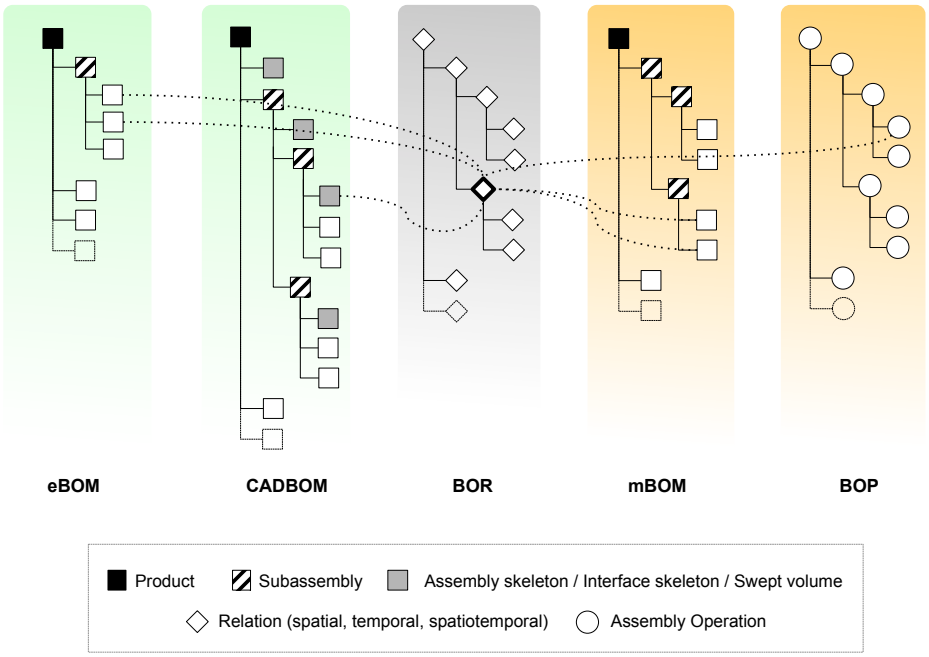

Figure 7: Introduction of the BOR as a reconciliation structure between PDM, MPM and CAD systems

Relation-based design considers the importance of relationships between

parts and assembly operations in order to increase efficiency in product design. The spatiotemporal relationships therefore give a great understanding of how parts are being connected in a dynamic way [3]. A relationship is here considered as the means to establish, represent or maintain a consistent link between two technical entities [9]. This one also gives information about product evolution, rationalises design intents, captures changes carried out over space and time and ensures the consistency of product-process definition [51]. In our context, mereotopological primitives and their ontological implementation enable the introduction of a spatiotemporal layer within PLM systems, by capturing five kinds of relationships and three kinds of association links as described below (the first two relations are reused from [9]) in Table 2.

Table 2: List of the relation types used in MERCURY

Relation type	Definition
Contact	Physical contact between two mechanical parts.
Precedence	Assembly logical order between two mechanical parts, which are (or not) in physical contact.
Design change	Additional information on intrinsic relation, which enables the spatiotemporal description of the change of size or change of form of the product (e.g. ***Growing***).
Kinematic pair positioning	Additional information on contact relation, which enables the spatiotemporal description of the positioning of a part on another one according to the defined degree of freedom (e.g. ***RevoluteOP***).
Technological pair positioning	Additional information on contact relation, which enables the spatiotemporal description of the positioning of a part on another to assume the part function (e.g. ***Arc Welding***).
Spatial link	Reference link between a part and its skeletons in product domain.
Temporal link	Link between two assembly operations in assembly process domain.
Spatiotemporal link	Parent-child link between two entities (i.e. a part and a swept volume) in the interface domain.

Figure 8 shows an emergent shift in information systems implementation. Actually, current implemented PLM systems in industry are generally

related to each other in pairs in order to exchange information [5]. It should be noted that system-to-system connections works in a non-reciprocal manner (e.g. PDM to MPM and CAD to MPM in Figure 8) and therefore provide weak interoperability capacity [32]. Here the management of heterogeneous information highlights the need of a seamless connection between information systems [32]. As a consequence, a hub-and-spokes architecture is proposed so as to ensure the orchestration of information flows and solve interoperability issues (right part of the Figure 8). The hub aims at introducing a top-management level to manage new technical objects and their inferred relationships (especially in the spatiotemporal dimension), and ensuring a better coordination of the PLM systems in the architectural design stage. The final objective is to enhance a continuity of product description/definition over space and time by introducing a novel spatiotemporal management framework.

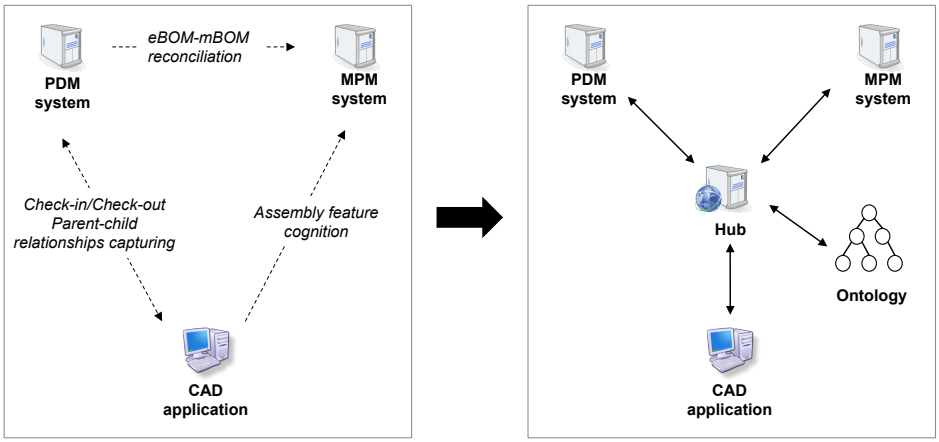

Figure 8: Shift in PLM systems implementation from current industrial usage to a hub-and-spokes architecture

The AOD philosophy requires the association and integration of specific viewpoints related to product design and assembly process planning domains. As such, the proposed efforts are based on the MUVOA model [9], which aims to separate and link concerns of both domains. Built on this, Figure 9 presents the relationships (intra-domain and inter-domain) between the views (e.g. structural, functional, contextual and so on) in the product design and assembly process domain, and also between both domains via a novel domain, the interface domain (related to the spatiotem-

poral dimension). In this specific domain, three views have been introduced (namely contextual, structural and geometric views) in order to describe the BOR, swept volumes and spatiotemporal relationships to name a few. For instance, this framework proposes to link the contextual view of the product domain and the contextual view in the assembly process domain through an added contextual view in the interface domain so as to promote information flow continuity in AOD. Table 3 shows the different views within their domains and the involved stakeholders. It should be noted that a novel actor has been introduced in order to cover the interface domain, the mediator. An example of technical objects for each view is also introduced.

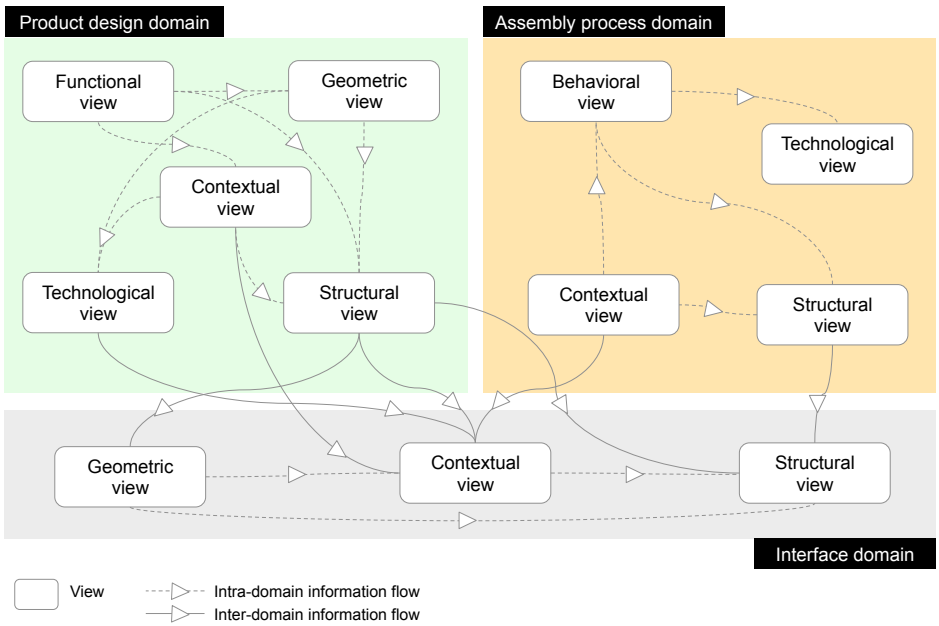

Figure 9: Adapted MUVOA model linking product design and assembly process domains through the interface domain

4.2. Information flow through PLM

This section introduces the information flow through PLM systems. In Figure 10, the hub is linked to PRONOIA2 ontology with a double arrow, which ensures the interoperability between systems by providing semantics among PDM, MPM and CAD systems. Semantics provides a common vocabulary between stakeholders involved in a project and gives via ontology

Table 3: Link between stakeholders and views in product and assembly process

Domain	View	Stakeholder	Technical object
Product	Functional	Product architect	Technical function
	Technological	Product architect & Designer	Technological pair
	Structural	Product architect	Product configuration, eBOM
	Geometric	Designer	Assembly constraint, part geometry
	Contextual	Assembly planner & Product architect	Kinematic pair, contact relation
Assembly process	Structural	Assembly planner	mBOM and BOP
	Contextual	Assembly planner	Precedence constraints and production context
	Behavioural	Assembly planner	Assembly sequence and process plan
	Technological	Assembly planner	Technological pairs and assembly technologies
Interface	Structural	Mediator	BOR
	Geometric	Mediator	Assembly and interface skeleton, Swept volume
	Contextual	Mediator	Kinematic & technological pair positioning & Design change (e.g. Union, Growing and Deformation)

a support to knowledge management. Besides, double arrows are present at each link in order to ensure seamless data changes at any time and information flow in a bidirectional manner. As such, bridges enable the importation and exportation of information from one information system to another. The hub is therefore considered as the dashboard, which controls information flow between engineering and manufacturing.

The entities, which are defined and managed within PLM systems (i.e. PDM and MPM systems) and CAD application, are also presented in Figure 10. Dotted arrows represent information as data input the hub, and full arrows are dedicated to information generated from information systems and applications. The PDM system exports the eBOM defined by the product architect and receives the restructured eBOM based on the generated assembly sequence. The MPM system receives the mBOM after that the assembly sequence has been defined by ASDA algorithm and exports the temporal regions and primitives. As for CAD application receives the CAD BOM and sends back the geometric information (such as geometric skeletons), as well as the changes made by the designer along the design process. All steps of Figure 10 are described in the upcoming paragraphs.

The hub – called Pegasus [7] – has been developed with Rhapsody software, which uses graphical models to generate software applications. Rhapsody facilitates collaboration and supports information tracking [23]. This modelling environment based on UML (Unified Modelling Language) has been used to assist stakeholders in managing information through visualisation. The proposed hub is a web-based system and a mediator application between information systems (i.e. PDM and MPM) and CAD tools in AOD [7]. Its role is to manage product relationships and control information (with spatial, temporal and spatiotemporal dimensions) flow between information systems and ontology with internal regulation procedures, so as to bridge the heterogeneity gap between engineering and manufacturing. Moreover, Pegasus provides a decision-making support for sub-assemblies (SA) selection and assembly operation structures generation by capturing design for assembly (DFA) rules. So Pegasus' purpose is to bring concurrent engineering benefits into the product design and assembly sequence planning stages. This hub has been improved in order to consider the new functionalities and new entities to be managed. Table 4 specifies the eleven functions implemented in Pegasus and the associated stakeholders. Here the MERCURY framework starts at the architectural design phase, by considering as data input defined functions and preliminary solutions. The proposed framework ensures the orchestration of information flows, reasoning capacity and then product-process reconciliation from a central place. The different steps of

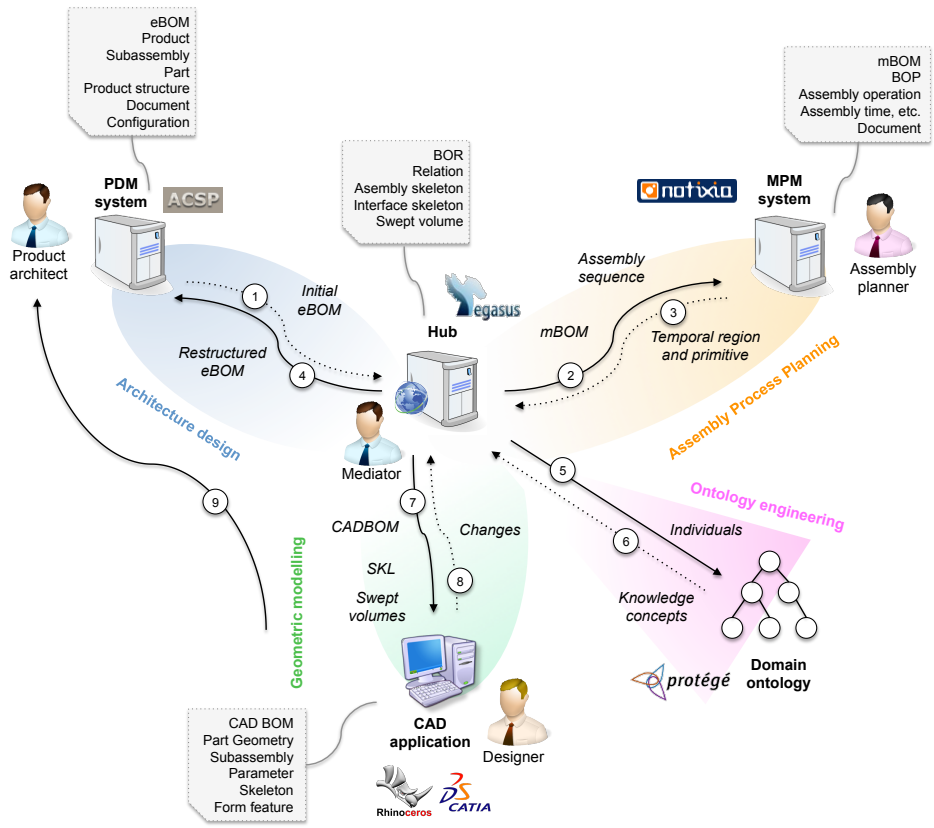

Figure 10: Main orchestration and reconciliation mechanisms within MERCURY and Pegasus

MERCURY framework can be deployed as follows:

Step 1 The product architect defines the eBOM in the PDM system. Then, Pegasus is automatically and directly populated with the spatial information from PDM. Spatial information corresponds to spatial region (e.g. part). The product architect also provides the part-to-part relationships within a graph (e.g. precedence, contact, kinematic pairs and technological pairs) in Pegasus. Spatiotemporal regions (i.e. parts which are going to move to be positioned in their final state) and kinematic pairs positioning are then computed from this aforementioned information.

Step 2 Based on the part-to-part relationships, the ASDA algorithm computes several admissible assembly sequences, on which the assembly planner will select the most appropriate one. Built on this, the mBOM is automatically exported to the MPM system, where assembly operations are defined from a technological perspective. As for the assembly sequence, operation-to-operation relationships are refined by using ***Precedence*** and temporal primitives.

Step 3 Manufacturing relational information is then sent to Pegasus.

Step 4 The eBOM, built on the assembly sequence and its related subassembly levels, is updated to the PDM system.

Step 5 All engineering and manufacturing data are sent to PRONOIA2 ontology so as to populate the application-ontology layer. Spatial, temporal and spatiotemporal information is automatically sorted out in the conceptual model (i.e. PRONOIA2 ontology), which is the top layer of the proposed framework.

Step 6 Then Pegasus synchronises the information from PDM and MPM systems through the PRONOIA2 ontology. Here the BOR is generated based on the previoulsy generated kinematic pairs positioning.

Step 7 The spatiotemporal graph of assembly evolution is generated from the eBOM, mBOM, swept volumes, kinematic pairs and temporal primitives within Pegasus. This enable the definition of assembly, interface skeletons as well as the CAD BOM and the simplified swept volumes in the CAD application. By considering this AOD context, the designer can build parts' geometry.

Step 8 Based on the assembly skeleton structure, the designer can allocate volume, shape and values related to each skeleton. The relational geometric information is then sent to Pegasus, which will capture the design and assembly intents. In addition, the designer has the opportunity to undertake changes, which are managed by automatically updating the BOR. Swept volumes or changed parts are dynamically represented and designers can directly see the impacts of changes. If other changes are required, then new information is again updated. Changes are captured in the spatiotemporal graph of design evolution, computed from changes listed by the designer along the design process. Changes can be visualised and dynamically carried out.

Table 4: Usage functions implemented in Pegasus and their related stakeholders

Product architect	Use functional requirements Define geometric requirements Define part-to-part relationships Define product structure Define assembly and interface skeletons
Assembly planner	Define manufacturing context Generate admissible assembly sequences Select sub-assemblies Evaluate sub-assemblies Evaluate admissible assembly sequences
Designer	Define design changes

Step 9 Once designer has defined all product geometric characteristics, the CAD models can be stored by using check-in/check-out procedures in the PDM system.

If a part is missing or the designer wants to change something, then he can restart the whole procedure in order to visualise new results. Moreover, the Pegasus application has been updated by integrating a domain-ontology and an application-ontology and representation means. Pegasus plays now a role of a semantic Web-service. An API (Application Programming Interface) has been created to extract relevant information from PDM and MPM systems and automaticall transfer it into the data model of Pegasus (developed with Rhapsody IBM software). Since the application-ontology does not use the same IT language as PLM systems and as they do not deal with the same information, the information needs to be converted with developed plug-in to be accessible to PRONOIA2 ontology. The inference reasoner (here HermiT 1.3.7) of the ontology processes checking procedure and ensures the consistency of the model. The ontology also enables structuring information and bringing semantics and logics in PLM systems. Pegasus and PRONOIA2 ontology (i.e. conceptual model) ensure interoperability and consistency between information systems and applications covering engineering and manufacturing departements with real-time information flow. Figure 11 presents how ontological information is integrated into information systems and application. Indeed, PRONOIA2 ontology is composed of six relevant parts (such as temporal primitives, spatial primitives, spatiotemporal primitives, spatial region, temporal region and spatiotemporal

region), such as represented in Figure 3. Here, spatial regions are defined in the PDM system, temporal regions in the MPM system and remaining information within the hub and PRONOIA2 ontology.

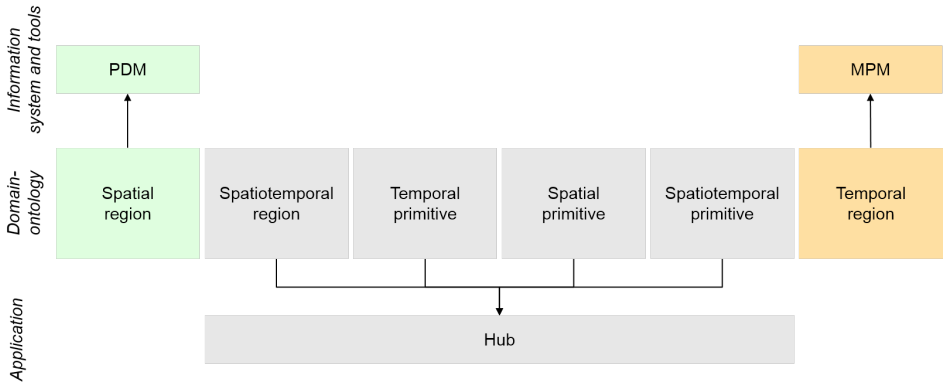

Figure 11: Ontological information integrated into PDM and MPM systems and the hub application

Moreover, Table 5 introduces the novel functionalities and the associated mereotopological concepts (such as proposed in [19]) embedded in Pegasus. In this table, the novel information is described in the second row, the input/output information (i.e. information imported from one system and exported to PRONOIA2 ontology through the hub) in the third row and the information tracking is shown in the last row. It should be noticed that the JANUS mereotopological theory has been entirely used within the hub so as to carry out its functionalities.

Figure 12 and Figure 13 have been introduced so as to describe MERCURY framework procedures about assembly and design evolutions. The proposed flowcharts present mechanisms (i.e. grey boxes) and related input/output information (i.e. white boxes). These mechanisms match the steps 1, 6, 7 and 8 of Figure 10. Figure 12 and Figure 13 focus on the assembly evolution application and design evolution application respectively. Here, the paper contributes to the generation of a BOR bridging spatial and temporal information, a CAD BOM, swept volumes and two spatiotemporal graphs. More precisely, the BOR links PDM and MPM heterogeneous information with dedicated relationships, introduced in the JANUS theory [19], such as kinematic pairs positioning and design changes. As such, Pegasus behaves like a mediator. Besides, the assembly and interface skeletons

Table 5: Functionalities and mereotopological concepts of the hub

	Functionalities	Mereotopological concepts
Information	Definition of contact relationship	Spatial primitive
	Definition of precedence relationship	Temporal primitive
	Definition of kinematic pair	Spatial primitive
	Creation of kinematic pair positioning	Spatiotemporal primitive
	Definition of technological pair	Spatiotemporal primitive
	Definition of design changes relationship	Spatiotemporal primitive
	Definition of temporal primitive	Temporal primitive
	Creation of swept volume	Spatiotemporal region
Input\Output information	Import and export of eBOM	Spatial region
	Import and export of spatiotemporal primitive	Spatiotemporal primitive
	Import and export of assembly operation	Temporal region
Information traceability	Representation of part-to-part relationships graph	Spatial region, temporal and spatiotemporal primitives
	Representation of purely spatial graph	Spatial region and spatiotemporal primitive
	Representation of the BOR	Spatial, temporal and spatiotemporal regions and spatiotemporal primitive
	Representation of assembly evolution graph	Spatial, temporal and spatiotemporal regions, and temporal and spatiotemporal primitives
	Representation of change definition graph	Spatial and temporal regions, and temporal and spatiotemporal primitives
	Representation of design evolution graph	Spatial, temporal and spatiotemporal regions, and temporal and spatiotemporal primitives

are defined in the JANUS theory with the description of spatiotemporal relationships. In addition, the related swept volumes provides a better understanding about how parts are being assembled at the early design stages. This also brings awareness to designer on how parts move during the assembly. At the end of Step 7, if the designer does not need to modify the product, the flowchart is over, otherwise the designer can keep on with step 8.

At the step 7 of the framework, a novel graph (such as introduced in [19]) has been introduced to support product evolution in the context of AOD based on assembly planning. Leonardo Da Vinci [11] states that "a picture is worth a thousand words". Actually, designers will be more aware of changes with a graph, which can visually explain an idea by outlining parts and relationships between them. The emergence process has already been followed in order to succeed to develop representations to designers. This leads to knowledge acquisition [17]. This graph is required, as current CAD trees do not introduce temporal and spatiotemporal information. They just show the history of the model building at the final design phase. As such they deal with static and purely spatial information. Although, this information is not enough to understand why the product is the way it is at the end of the design process. The spatiotemporal graph gives to product architect and designer the opportunity to have an overall view of the product assembly evolution by representing milestones in collaborative project. Indeed it federates, represents and links spatial and temporal information. With such a graph, stakeholders can understand how entities are linked over space and time, and how they interact. As such, the information propagation through design and manufacturing structures is managed. Table 6 introduces the link between the new views of the MUVOA model (see Figure 9) and the MERCURY framework. This table highlights the reasons why the MUVOA model has been improved.

To sum up, MERCURY has been inspired by the musical domain. An orchestra is for instance composed of wind (here PDM system) and brass (here MPM system) instruments. The orchestra is guided by the conductor (here the hub), which gives the rhythm. Each family of instruments (here information systems) includes musicians (here agents). Each musician will make his own sound (here link between the information systems and hub), which will be heard by the conductor. All sounds, coming from wind and brass instruments, create a sheet music (here the ontology is instantiated). If music is created by all instruments family, then it is a full symphony.

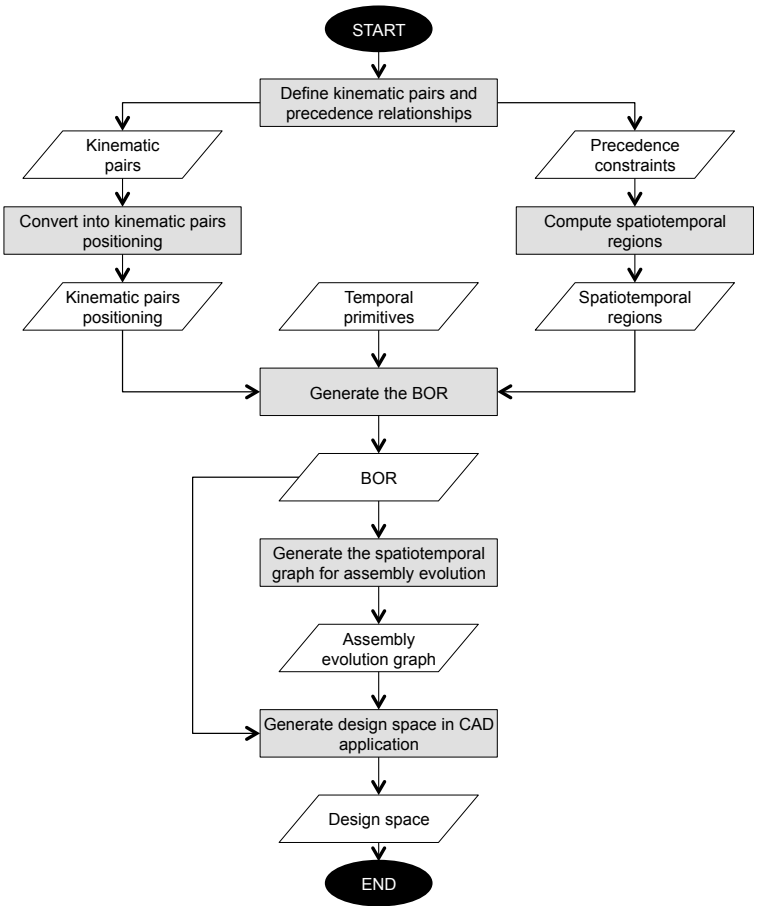

Figure 12: Flowchart of the proposed framework to manage “assembly evolution” within Pegasus hub

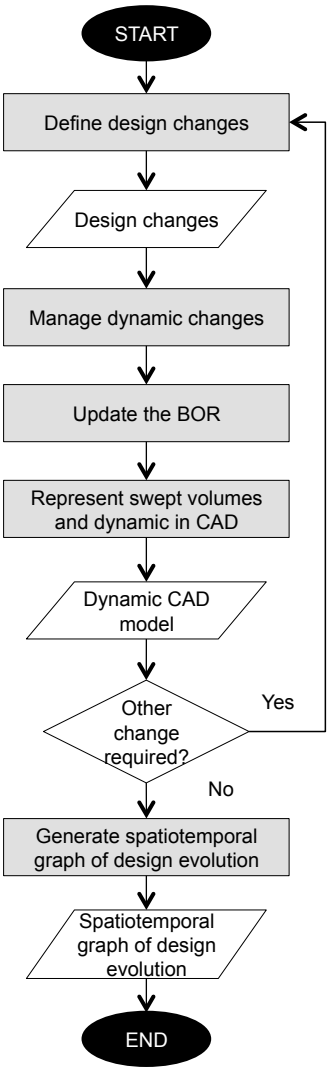

Figure 13: Flowchart of the proposed framework to manage "design evolution" within Pegasus hub

Table 6: Link between new product-process views of the MUVOA model and the MERCURY framework

View	Related step number	Reason for this improvement
Geometric	1	Need of managing new entities describing evolutions during the assembly and design process
Contextual	7 and 8	Need of new tools enabling describing product evolution during the assembly process so as to keep tracking of product architect's intents, as well as need of new tools enabling changes description so as to keep a track of designer's intents
Structural	6	Need of a new structure for design information based on relationships, as new spatiotemporal entities are used and required to be integrated

4.3. Representation of design and assembly evolutions within a CAD application

Current CAD tools (e.g. Catia V5) can only design static spatial product[2] and represent the final state of the physical model [13]. They are considered as endurant software, which is currently their major drawback [25]. In real world, products are dynamically evolving over space and time. As such, Rhinoceros software[1] (i.e. perdurant software) with its Grasshopper[2] plug-in, which are open-source, has been exploited, so as to instantaneously visualize assembly and design evolutions of products. This visual programming tool can generate geometry with interactive modification [6]. The overall objective of using Grasshopper is for the designer to control all evolutions that the product could encounter during the early product development process. By doing so, JANUS mereotopological theory has been implemented within a CAD tool (Rhinoceros) and via dedicated algorithmic procedures (similar to building blocks) developed within Grasshopper (see example in

[1] https://www.rhino3d.com/en
[2] http://www.grasshopper3d.com

Figure 14).

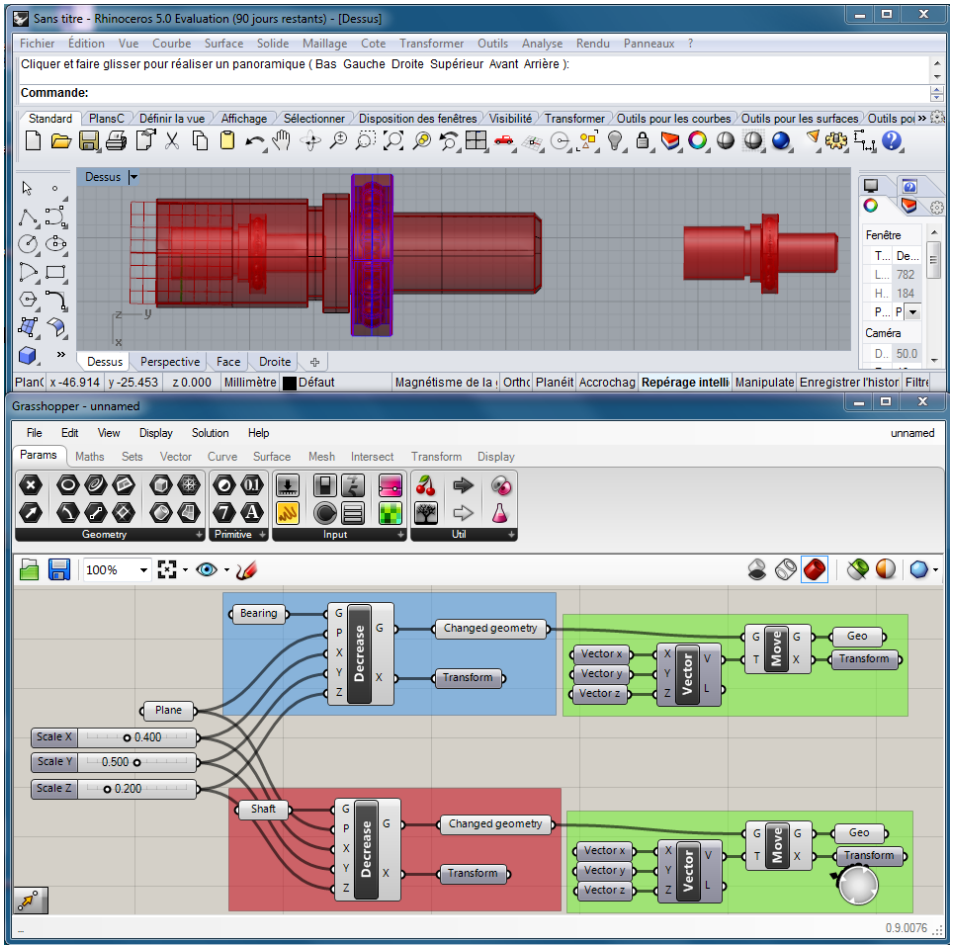

Figure 14: Dynamic representation of ***Decrease*** relationship within Rhinoceros and Grasshopper

Figure 14 presents an example of design evolution, namely the ***Decrease*** relationship. This product has been first designed in Catia V5 and then transferred to Rhinoceros. Here, the red part needs to be rescaled and decreased because of the designers experience. The designer just uses the developed boxes in Grasshopper (e.g. Decrease box) and links it to the changing parts in Rhinoceros (e.g. link with the Shaft). The designer dynamically and separately changes the x,y,z dimensions of the parts. Here, the parts have been decreased of 60% in the x dimension, 50% in the y

dimension and 80% in the z dimension. After choosing the right length and width, a smaller product is obtained, controlled by the red box for the shaft and the blue box for the bearing. The smaller product has been moved to the right side of the figure so as to have a better visualization. Here the shaft and the bearing have been linked through scaling parameters. As such, the relationships between these two parts is ensured over time even when the product undergoes changes. Figure 15 represents the swept volume (i.e. in red) of the bearing, resulting from the ***RevoluteOP*** primitive. A box named "swept volume" has been developed with Grasshopper in order to visualize the move of the part on the product. As such, swept volume is automatically created after defining the start and end interface (based on interface skeletons previously described in JANUS theory), as well as the directed curve of the move (based on the assembly skeletons). With such a representation, designer can dynamically see parts' moves (from a starting interface to the final position in the product assembly) and the space needed during the assembly process.

Figure 16 presents the creation of the swept volume of the ***CylindricalOP*** from step 1 to step 4. The different pictures below explain the main steps of the swept volume creation within CAD tools. For the simplified swept volume generation, interface skeleton is extended. On the contrary here in step 4, *Part 5* is extended to generate *Swept volume 5*. The figure also compares the reasoning of the theory (from left to right on the chronological order) and within CAD tools (from right to left). Some similarities appear between the description of the theory and the one of the CAD tool. In both descriptions, skeletons are used to generate the swept volume. In CAD tool, the final state of the assembly with its skeleton is considered and the path of the part, as well as its swept volume afterwards. In the JANUS theory, discrete parts are considered and the swept volume is created when a part is moving to get to its position in the assembly. In future works, all kinematic pairs, technological pairs and design changes should be represented with Grasshopper. The main advantage of this plug-in is that the change, defined at the step 8 of the framework (cf. Figure 10), is instantaneously represented and the designer can dynamically play with it. As such, design process is no more static and can now be considered as dynamic with product design evolution over time.

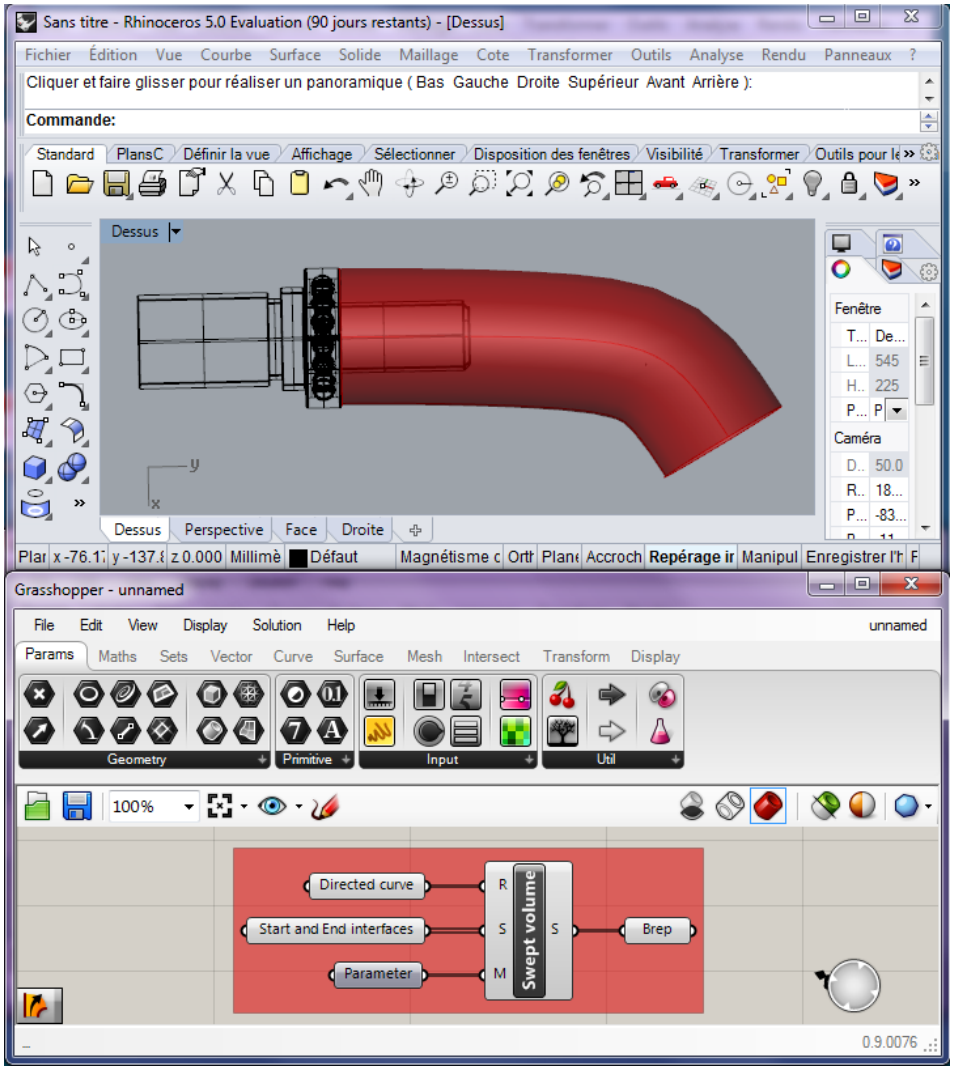

Figure 15: Dynamic representation of the swept volume resulting from the ***Revolute*** relationship within Rhinoceros and Grasshopper

5. Case study in a drones-based application

5.1. Case study: Drones-based design for medical emergency

The case study is solved by following the different steps of the proposed framework (cf. Figure 10). The objective of this case study is to design a medical drone. Medical drone is mainly used to reduce the delay between the

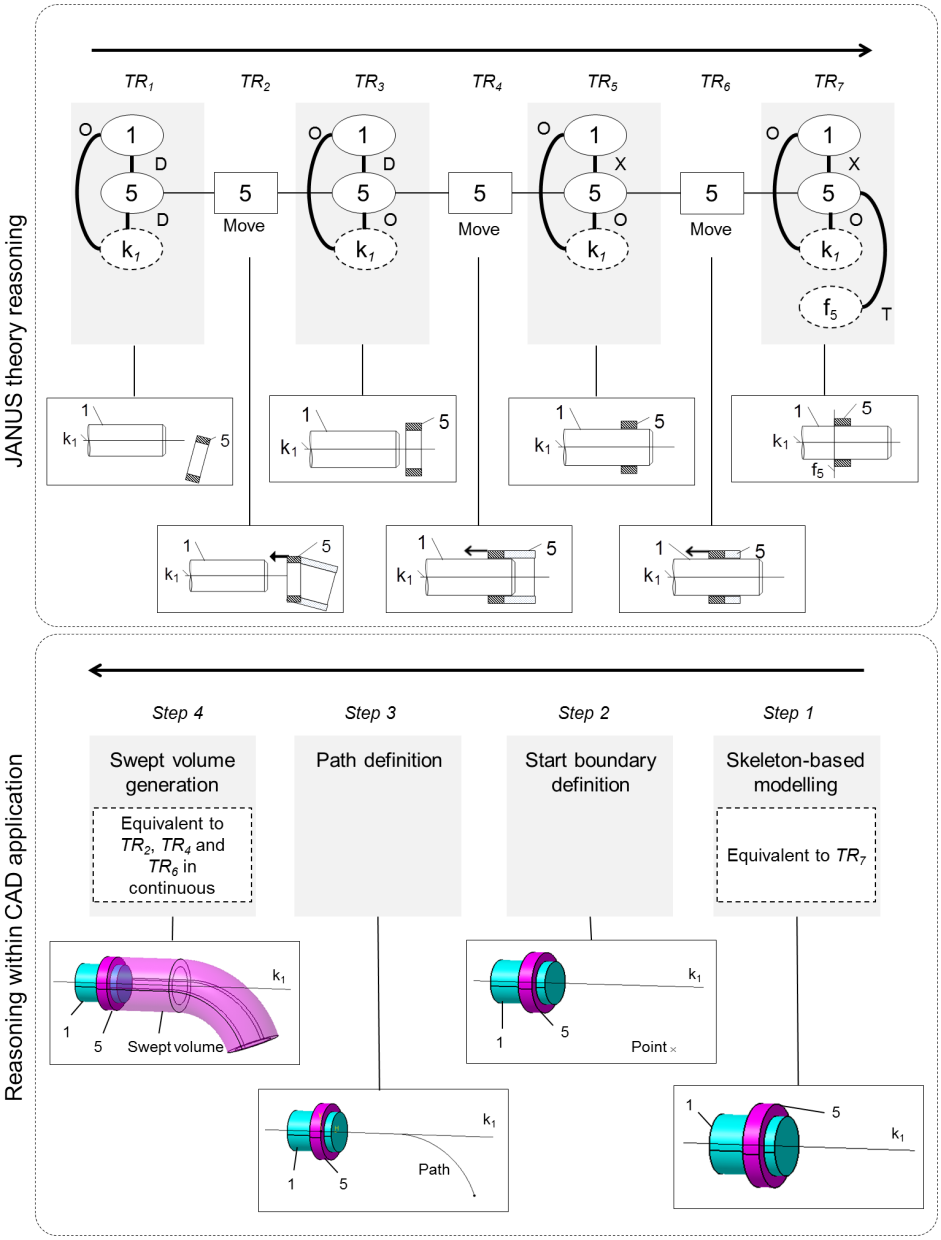

Figure 16: Comparison between JANUS theory and CAD application reasoning on assembly evolution

alert time and the intervention time. This time can actually be crucial for heart attacks, research of victims and so on. First, functional analysis has been carried out in order to understand the needs and find some preliminary solutions (cf. Table 7). After defining the requirements, some pre-concepts have been drawn. Based on morphogenesis, one solution is illustrated in Figure 17. Here, the case study has been simplified and only described a part of the whole assembly. The case study border has been marked out in Figure 17 with a black circle.

Table 7: Functions and related solutions for the case study

Functions	Solutions
Fly in order to bring medical equipment	Four drive turbines
Land on every kind of grounds	Landing gear with three legs
Go anywhere	Retractable rear arms

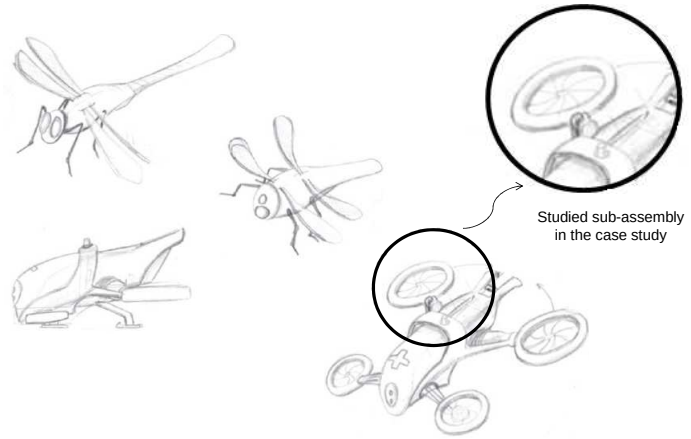

Figure 17: Representation of a pre-concept using morphogenesis

5.2. MERCURY deployment regarding assembly evolution

Define eBOM The eBOM is composed of ten parts, represented in ACSP PDM system on Figure 18 and by circles on Figure 19.

Define kinematic pairs and precedence relationships The part-to-part relationships are defined in a directed graph (cf. Figure 19) by the

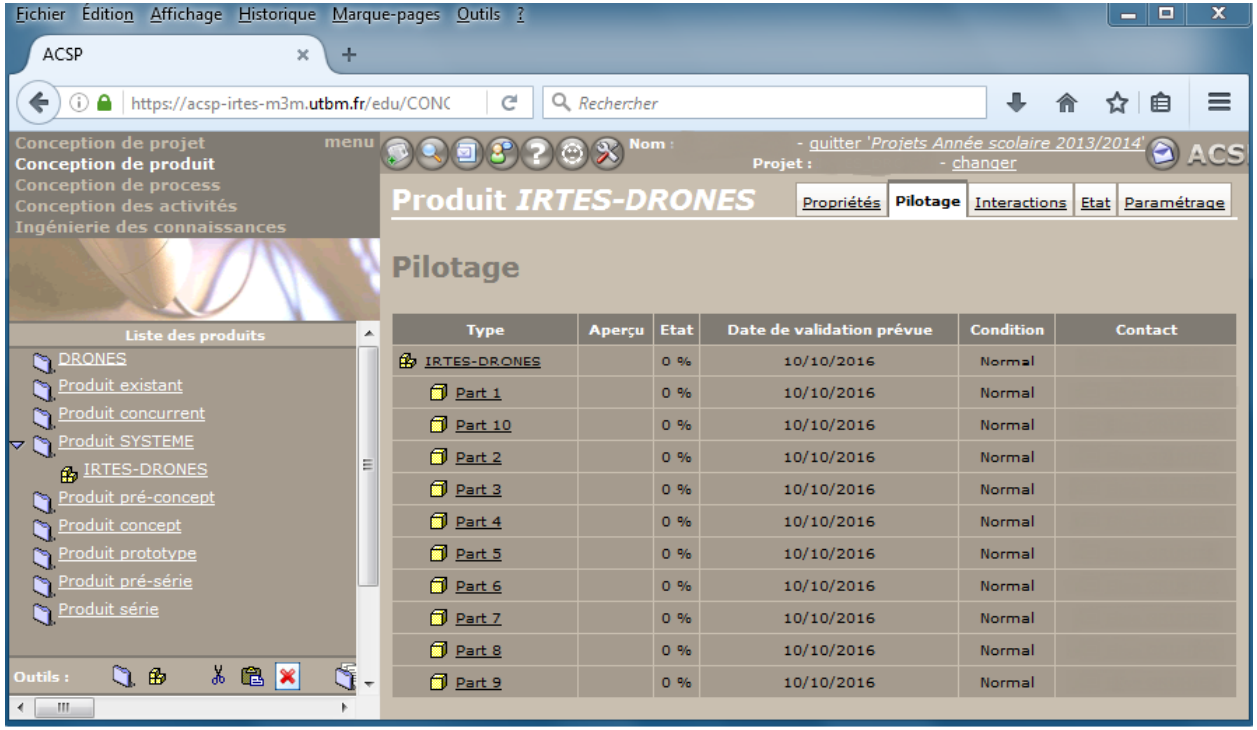

Figure 18: eBOM defined in ACSP as PDM system

product architect representing the contextual view of the proposed model. In this graph, the contact (i.e. link between two circles), precedence relationships (i.e. arrow) and kinematic pairs between two parts can be visualised. At the same time, a purely spatial graph is created in the right side of Figure 19. This graph represents the final state of the assembly process. With such a graph, stakeholders can know for instance if a part is at the core of the assembly. The green indicator means that steps have already been done, orange indicator means "work in progress" and red means that works need to be done later.

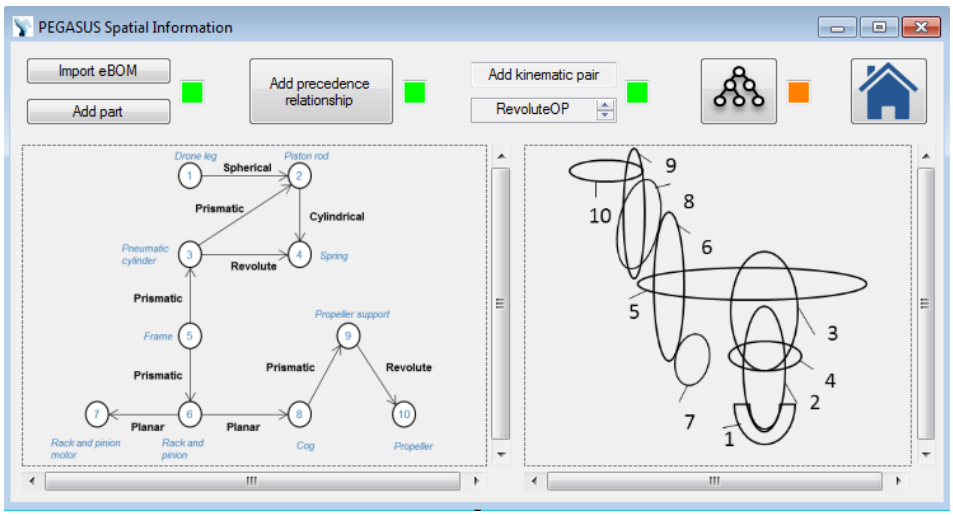

Figure 19: Contextual view of Pegasus hub

Deduce spatiotemporal regions and convert kinematic pairs positioning

Each spatiotemporal region is deduced from the part-to-part relationships graph. In fact, spatiotemporal regions are presented as the parts touching the end of the arrow in Figure 19. Here the swept volumes (i.e. spatiotemporal regions) are *Swept volume 2*, *Swept volume 4*, *Swept volume 7*, *Swept volume 9*, *Swept volume 10*, *Swept volume SA1*, *Swept volume SA3* and *Swept volume SA4*. Kinematic pairs positioning are also directly generated from the defined kinematic pairs. For instance, Revolute pair is converted into ***RevoluteOP*** with the JANUS theory. The parts and skeletons are listed in Table 8, where k and f are respectively the assembly and interface skeletons. In brackets,

Table 8: Parts list of the case study

No	Part name
1	Drone leg
2	Piston rod
3	Pneumatic cylinder
4	Spring
5	Frame
6	Rack and pinion
7	Rack and pinion motor
8	Cog
9	Propeller support
10	Propeller

Table 9: Assembly skeletons and interface skeletons list of the case study

No	Assembly skeleton	No	Interface skeleton
k_1	Line (4-2)	f_1	Surface (4-2)
k_2	Point (2-1)	f_2	Surface (2-1)
k_3	Plane (7-6)	f_3	Surface (7-6)
k_4	Plane (SA3-SA2)	f_4	Surface (SA3-SA2)
k_5	Line (SA4-5)	f_5	Surface (SA4-5)
k_6	Line (10-9)	f_6	Surface (10-9)
k_7	Plane (10-9)	f_8	Surface (9-8)
k_8	Line (9-8)	f_9	Surface (SA1-5)
k_9	Line (SA1-5)	f_{10}	Surface (4-3)
k_{10}	Plane (4-3)		

both parts and sub-assemblies linked with the skeleton are mentioned. These skeletons depends on the chosen spatiotemporal relationships and their related mereotopological description.

Generate assembly sequence Based on part-to-part relationships, the ASDA algorithm generates a relevant assembly sequence, which can be expressed as:

Assembly = $[SA1\ ;\ 5\ ;\ SA4]$
Assembly = $[[1\ ;\ 2\ ;\ 3\ ;\ 4]\ ;\ 5\ ;\ [SA3\ ;\ SA2\ ;\ 10]]$
Assembly = $[[1\ ;\ 2\ ;\ 3\ ;\ 4]\ ;\ 5\ ;\ [[8\ ;\ 9]\ ;\ [6\ ;\ 7]\ ;\ 10]]$

So the spatial regions *1, 2, 3* and *4* are embedded in a Sub-Assembly

(denoted *SA1*), which represents the landing system. Besides, the spatial regions *6* and *7* are embedded in a Sub-Assembly (denoted *SA2*) to represent the rack and pinion system and the spatial regions *8* and *9* are embedded in a Sub-Assembly (denoted *SA3*) for the arm system. Finally, the spatial regions *SA3, SA2* and *10* are embedded in a Sub-Assembly (denoted *SA4*). The assembly sequence is exported to MPM and then refined by the assembly planner. Operations are defined from the assembly sequence. Here, the different assembly operations are expressed within Notixia[3] MPM system (cf. Figure 20). The restructured eBOM (cf. Figure 21) considers the parts, as well as the sub-assemblies previously detected.

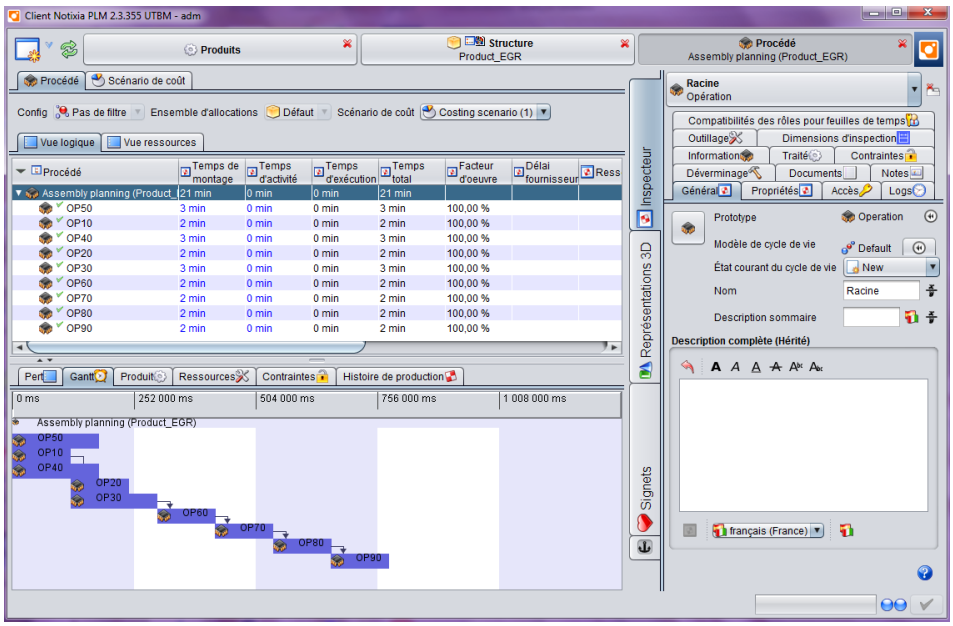

Figure 20: Representation of the assembly sequence planning within MPM system (i.e. Notixia)

For this case study, the third level of PRONOIA 2 has been automatically populated with previous information. The individuals are shown in Figure 22 and are linked together with properties. This figure is composed of nine states represented in the middle. The dif-

[3]http://www.witlaken.com/en/industrial/assembly-shop-solution/

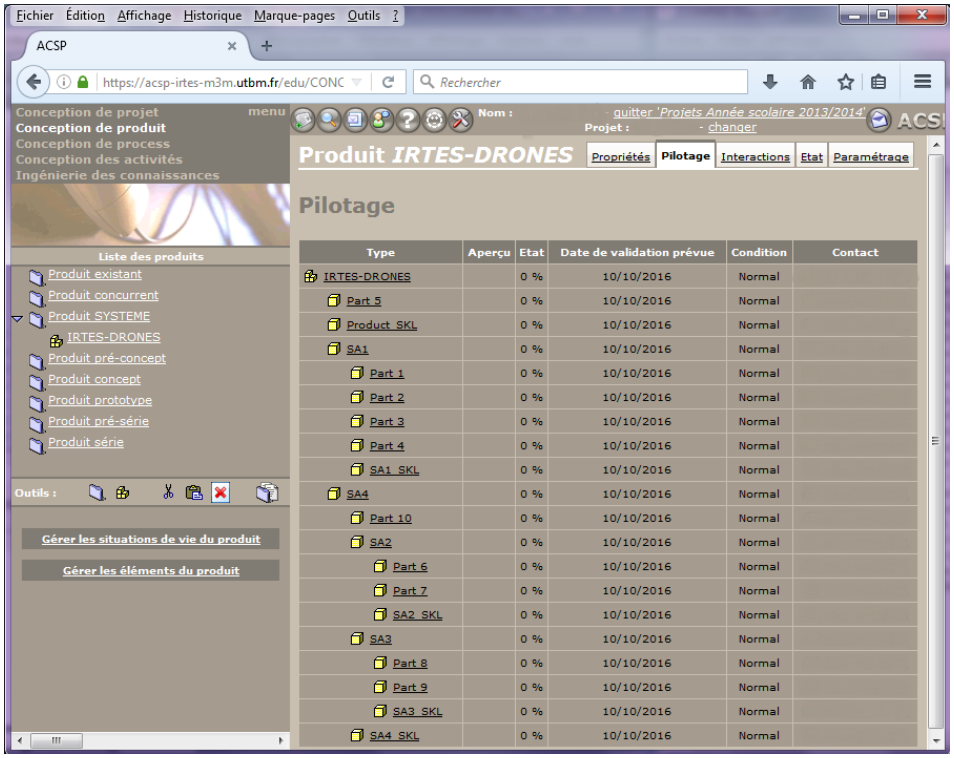

Figure 21: Restructured eBOM of the case study within PDM system (i.e. ASCP)

ferent states allow visualising what happens in the context of AOD and show the product evolution over space and time. After populating the ontology, designers can check that the application-ontology is composed of all concepts needed in the description of product-process evolution through AOD as well as their consistency. The ontology in its current shape proposes some available queries to designers so as to look for information requests in a collaborative assembly design environment. In addition, the reasoner can query the ontology and give designers suggestions on how parts will be assembled over time. So it could also provide some design recommendations to support designers. After having added information into PRONOIA2, designers have a suitable design context based on perdurantist vision to define consistent mechanical assembly.

Generate the BOR The BOR is generated from previous information:

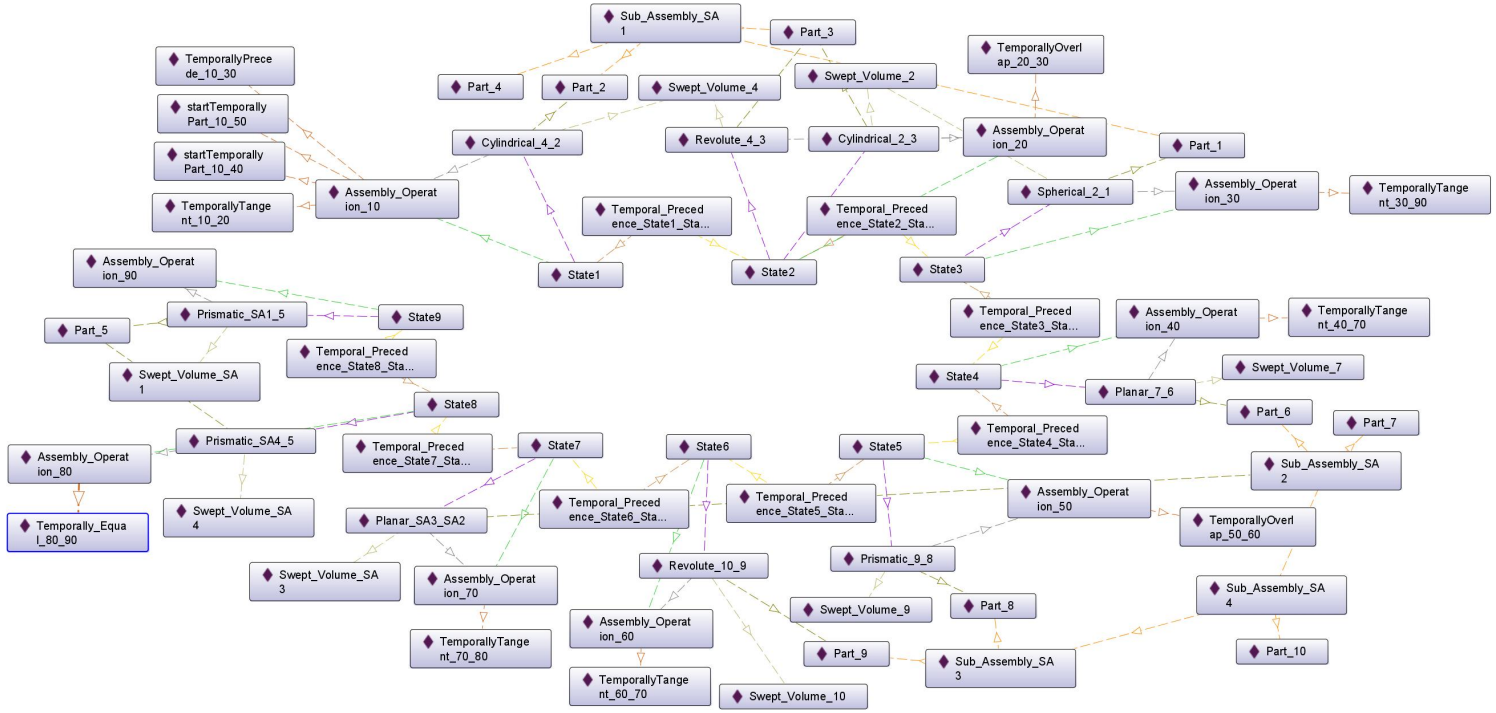

Figure 22: Instances of the ontology with the “kinematic pairs positioning” primitives for the case study

eBOM, BOP, swept volumes and kinematic pairs positioning.

Generate the spatiotemporal graph If the instantiated information in PRONOIA2 ontology is consistent, the "assembly evolution" spatiotemporal graph can be generated. All information is gathered to describe product evolution, horizontally read from left to right (cf. Figure 23). Hence, spatial (i.e. mechanical part), temporal (i.e. assembly operation) and spatiotemporal (i.e. swept volume in the context of AOD) regions are represented and linked with primitives. Only regions, which undergo a change at a specific assembly operation *OPi*, are presented at *OPi* in order to facilitate understanding of the reader. Here, changes are considered when novel spatiotemporal relationships appear. As such, the product architect and designer can understand what happens for each operation, especially where a part is situated and linked regarding to others, and which part is moving. In that case, the designer knows which part is evolving (and becomes a swept volume) and which part remains static. In fact, the designer can follow how each object is evolving over time and what are the changes that the product undergoes in the context of AOD. For instance, he can read, through the spatiotemporal graph of the hub application, that at *OP30*, *Swept volume 2* is moving to be positioned in a ***SphericalOP*** manner on *Part 1*. Then this representation, embedded in Pegasus hub, provides an overview of the product evolution along its assembly sequence to the product architect and designer before starting the product modelling phase in CAD. This representation therefore introduces a novel working support to the product architect and designer by giving them an overview of product-process evolution. Table 10 compares current CAD information and our proposed graph. Current CAD tools do not capture temporal (i.e. assembly sequence) and spatiotemporal information (i.e. spatial evolution of the product over time). Therefore, some information is missing in actual CAD tools.

Generate skeletons, simplified swept volumes, and CAD BOM Assembly and interface skeletons are generated based on the selected kinematic pairs positioning. For each spatiotemporal relationship, related skeletons have been defined in JANUS theory. The different skeletons are represented in Figure 24. Here the future CAD BOM has been rethought. The CAD BOM includes spatiotemporal information. From these relationships, specific skeletons (denoted SKL) and swept volumes (denoted SV) are created. This information has been included

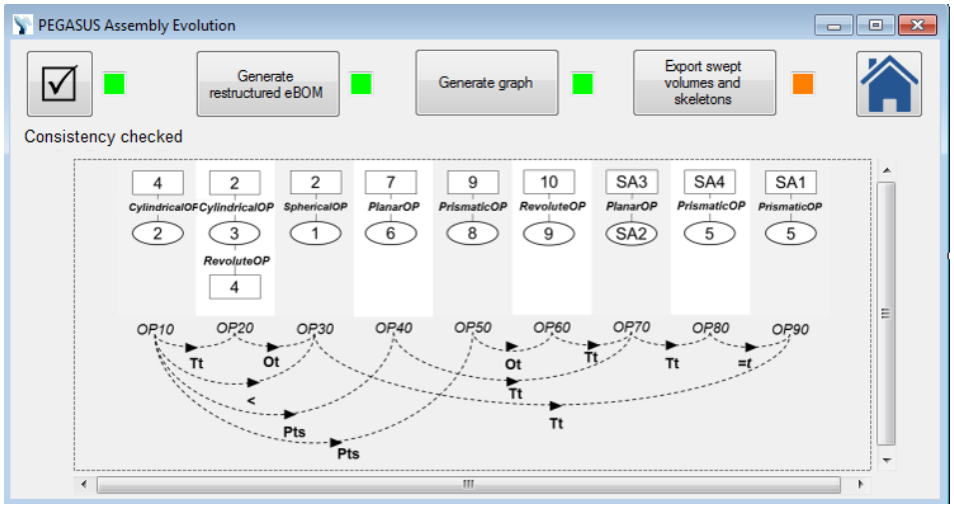

Figure 23: Geometric view of "Assembly evolution graph" Pegasus hub interface

Table 10: Comparison between current CAD information and our proposed graph

	Current CAD information	Proposed graph
Qualitative data	√	√
Quantitative data	√	
Spatial information	√	√
Temporal information		√
Spatiotemporal information		√

in the restructured eBOM, where designers develop parts. As such, these design activity is really based on relationships. The CAD BOM is exported to CAD tool based on previous information. Simplified swept volume is represented in Figure 25. They are created based on interface skeleton, a start point and the path. The red arrow represents the direction of the part move. Different types of simplified swept volumes exist such as cylindrical or parallelepipedal. These swept volumes bring information on how parts are moving during the assembly and can aid designers to design parts within a CAD tool. Besides, this kind of representation gives an overview to the product architect of designer's intents. A tracking of the product history is also kept for

future product development.

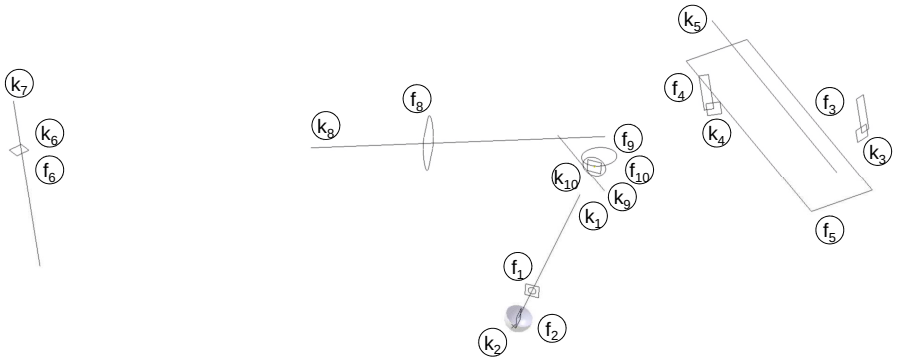

Figure 24: Assembly and interface skeletons of the case study

5.3. MERCURY deployment regarding design evolution

Create CAD model The designer creates the parts of the assembly using a CAD application (cf. Figure 26) based on assembly and interface skeletons, and simplified swept volumes and knowing that information from PDM and MPM systems is consistent.

Define design changes The designer's experience aids him to shortly notice if the product is going to resist deformation or can be simplified to earn time during the manufacturing process. As such, designers may want to change a relationship between parts or define a new one like "design changes" previously defined. For instance, here the designer chooses to increase the drone leg dimension to have a better support on the ground. Besides, a change of form of the drone leg in contact with the ground is required by the designer. These changes are manually entered inside the interface, like on Figure 27.

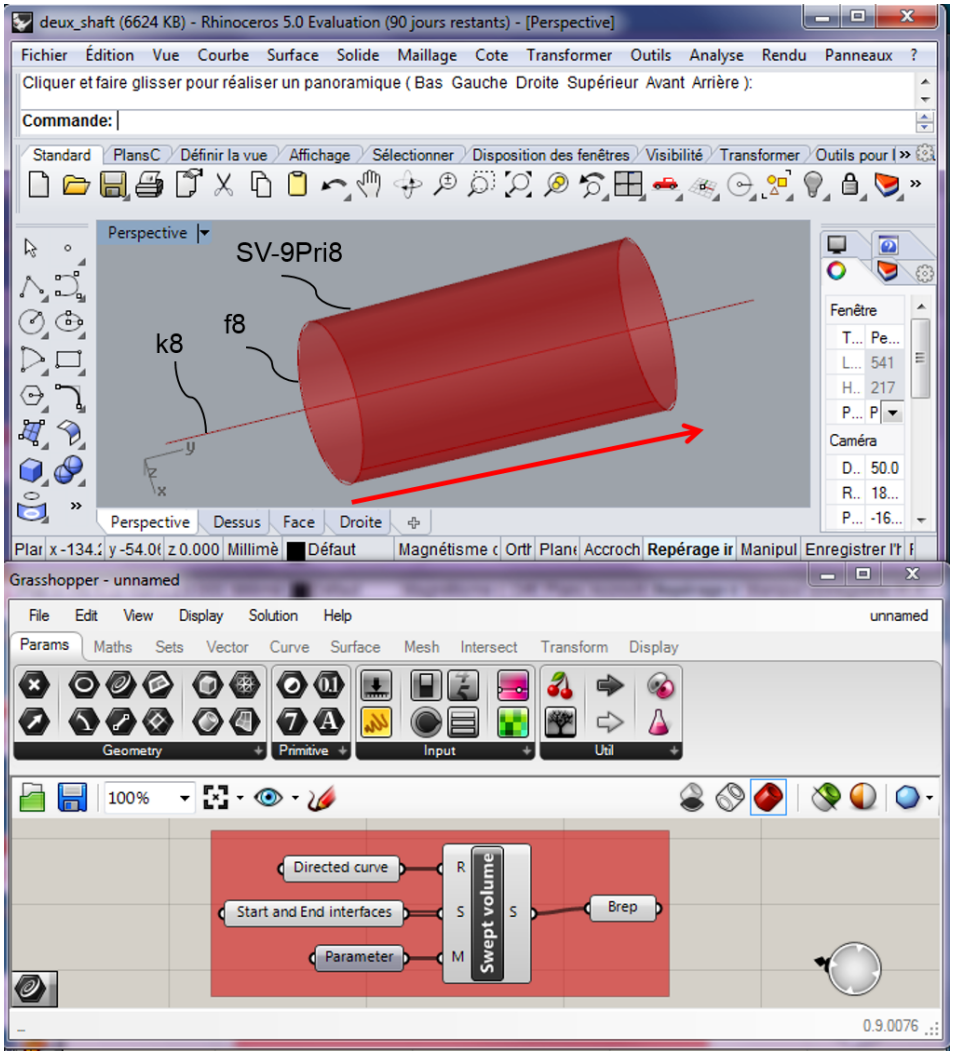

Figure 25: Simplified swept volume representation of the case study within Rhino/Grasshopper

Manage dynamic changes and update the BOR After defining what needs to be changed, the designer undertakes the modification in a CAD application, based on the imported BOR. The dynamic changes can be visualized within Grasshopper and instantaneously undergone. At the same time, the ontology (cf. Figure 28) organizes information about design changes in the hub application.

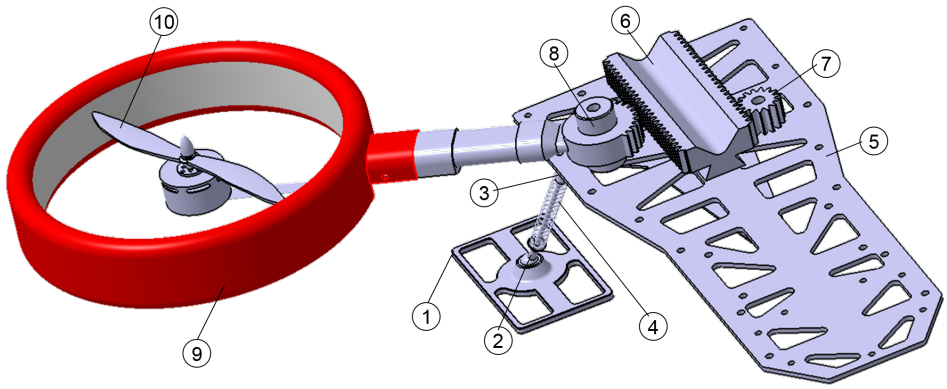

Figure 26: Studied assembly within Catia V5

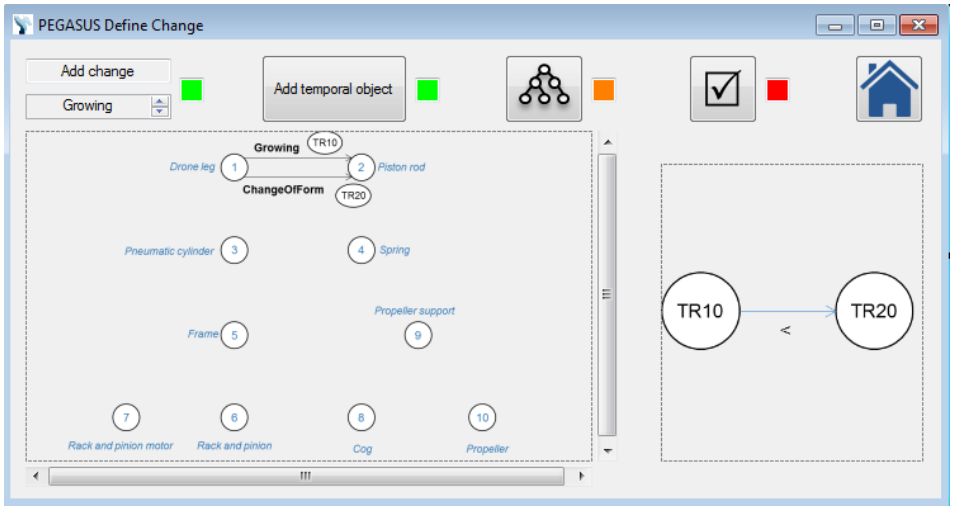

Figure 27: Contextual view of the "Design change" Pegasus hub interface

Generate the spatiotemporal graph Then, a spatiotemporal graph (cf. left side of Figure 29) is generated, which sums up all changes undertaken by the designer during the design process. The ***Growing*** primitive and its related swept volume (i.e. SV_1Gro2, standing for swept volume of the growth of Part 1) are introduced in the right side with the red arrow representing the direction of the growth. In fact, when the designer chooses to grow a part, the change is indicated within the hub, kept for tracking with the spatiotemporal graph and imple-

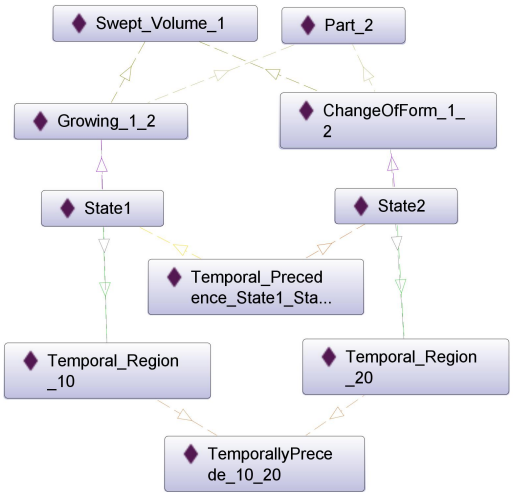

Figure 28: Instances of the application-ontology with the "design changes" primitives

mented in a CAD tool to visualise the related swept volume. This mechanical assembly can then be visualised within a CAD application as illustrated in Figure 30. The case study has only described a part of the whole assembly. Here, the parts *8, 9, 10* and the parts *1, 2, 3, 4*, such as represented in Figure 26, have been copied, since the product has a symmetry. The designer has also added a shell covering the whole system so as to have a fashionable and ergonomic product.

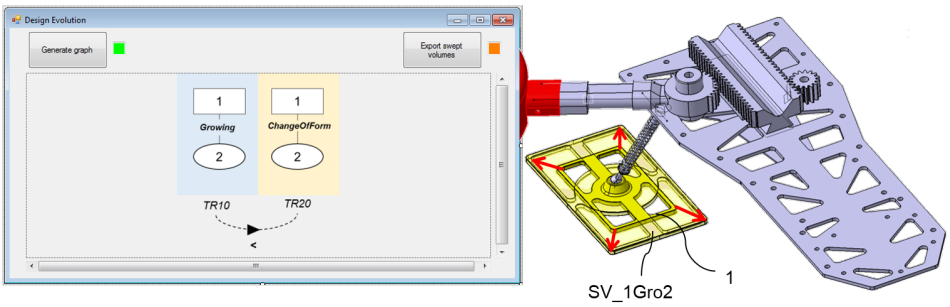

Figure 29: "Design evolution" interface of Pegasus hub and illustration of the ***Growing*** relationship with its related swept volume

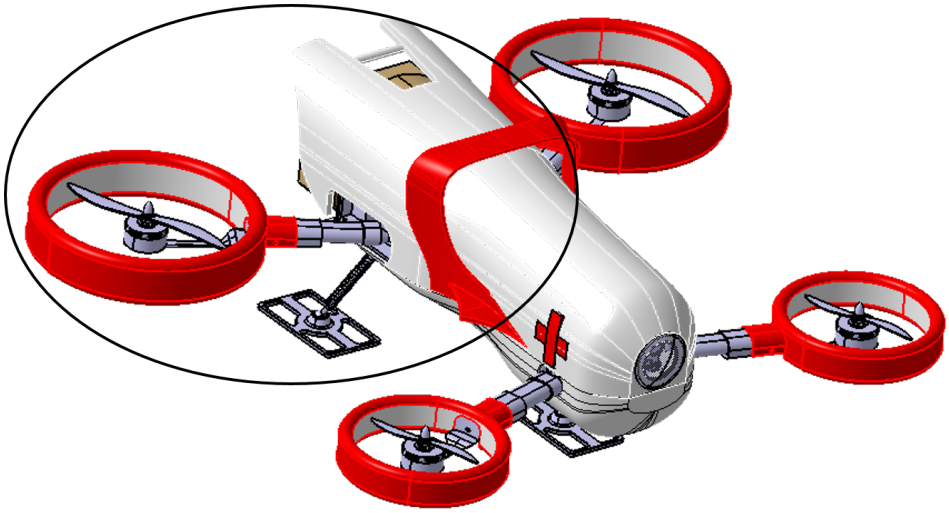

Figure 30: Representation of the whole product within Catia V5

The UML sequence diagram (cf. Figure 31) illustrates and sums up the information flow through tasks in Pegasus hub. Here the hub with the mediator are at the core of the product development process by bringing a novel semantic and logical layer to current design stages.

6. Discussions

In this paper, PRONOIA2 ontology based on spatiotemporal mereotopology has been integrated in a novel management framework and implemented into a hub-and-spokes application (Pegasus) and linked to PLM systems such as PDM, MPM and CAD. In a general way, the MERCURY framework enables the management of spatiotemporal information so as to ensure information flow continuity and consistency between engineering and manufacturing. However, it remains to be tested at the industrial level in order to get some industrial feedbacks, which could contribute to improve the research work. The proposed framework has only been validated with simple case studies including few parts, it will be important to deploy the efforts to large assemblies in a near future. Besides, more dynamic changes could be developed within Grasshopper so as to deliver a wider range of possible design changes to the product architect and designer. The link between Grasshopper and Catia could also be automated and made reversible, since it has been currently executed manually in the experimentation. Moreover,

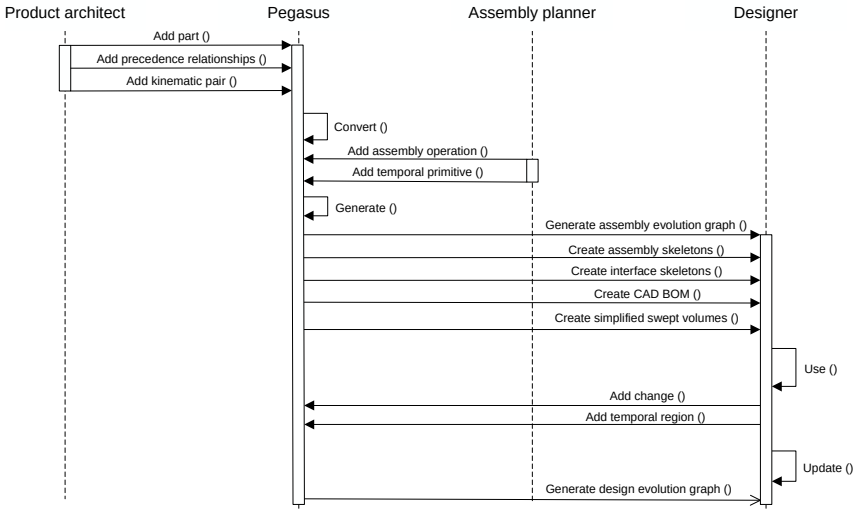

Figure 31: UML sequence diagram showing activities and the related stakeholders during the design process

simplified swept volumes could be automatically updated by considering an equivalent volume as soon as designer creates the part geometry. This updated equivalent volume would be closer to the real volume of the part at the end of the design process.

The framework (including the JANUS theory and PRONOIA2 ontology) can be enriched with an extension to other domains. Indeed, a reconciliation with other information systems should be addressed in order to describe evolution in other lifecycle domains. For instance, the theory could be adapted to describe the evolution in MOD (Manufacturing-Oriented Design) and new restrictions and rules could be added to the ontology to check the information consistency. Finally the ontology would be linked to new information systems (e.g. CAM tools), which match the new domain of research. As a result, the MERCURY framework could be adapted and applied to every domain where objects evolve over space and time and where heterogeneity issues persist.

7. Conclusions and future work

This paper has introduced an ontology-based spatiotemporal information management framework for product design and assembly planning reconcili-

ation. The heterogeneity issue raised by the integration of assembly process planning in product design (temporal information from MPM system integrated with spatial information from PDM system and CAD application) has been solved in the upstream design stages with the introduction of new technical entities (i.e. spatiotemporal relationships). Information has been qualitatively described in previous works with JANUS mereotopological theory, which captures the characteristics of the physical entities in the actual world. Here mereotopology can aid product architect and designer by describing where the different parts of the product are placed each other at different instants and what are the connections between them. The theory has then been formalised by PRONOIA2 ontology, which brings a semantic and logical layer to PLM procedures, as well as ensures information interoperability between information systems. Besides, Pegasus hub has been developed so as to store new entities (i.e. swept volume and spatiotemporal relationships to name a few) and formal changes description, not yet supported by current information systems. PRONOIA2 ontology also enables structuring the novel information in Pegasus hub.

The MERCURY framework has been developed to manage product design and assembly planning reconciliation, and orchestrate the information flows through associated information systems and tools. MERCURY enables bringing the spatiotemporal dimension within CAD tools by representing the assembly and design evolutions undergone by the product. The spatiotemporal aspect enables introducing dynamic within CAD tools so as to minimise design time. Indeed, information is added at the early design stages. Besides, designer can easily modify the product along the design process and changes can be instantaneously represented within Grasshopper. MERCURY also brings to product architect, assembly planner and designer a context, defined by relational information and common semantics. Indeed, the reasoner of the ontology enables checking the conceptual model consistency, on which information from PDM, MPM and CAD systems is based, as well as ensuring the development of a consistent product definition. The hub also keeps traceable change history such as assembly and design evolutions (useful for future designs) and provides a whole overview of product evolutions and changes throughout the product lifecycle with new graphs representing design intents. Therefore, the framework aims at promoting collaborative, proactive and concurrent engineering for the design of ever-evolving product.

In future work, the way of designing will be fully build in terms of relationships, so as to promote relational design. The dynamic aspect of CAD tools will be improved in order to support designers in the early design stages

through the integration of BOR and spatiotemporal features leading to a spatiotemporal CAD. Moreover, works initiated in the Grasshopper plug-in will be kept on so as to represent more and more evolutions encountered in the assembly and design processes.

References

[1] Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM 26*(11), 832–843.

[2] Bauzer Medeiros, C. (2002). Spatiotemporal information systems.

[3] Brown, J. (2006). Managing product relationships Enabling iteration and innovation in design. Technical report, AberdeenGroup.

[4] Chen, L. and H. Luo (2014). A BIM-based construction quality management model and its applications. *Automation in Construction 46*, 64–73.

[5] Ciocoiu, M., D. S. Nau, and M. Gruninger (2001). Ontologies for Integrating Engineering Applications. *Journal of Computing and Information Science in Engineering 1*(1), 12.

[6] David, M. and F. Rowe (2015). What does PLMS (product lifecycle management systems) manage data or documents complementarity and contingency for SMEs. *Computers in Industry*.

[7] Demoly, F., O. Dutartre, X.-T. Yan, B. Eynard, D. Kiritsis, and S. Gomes (2013). Product relationships management enabler for concurrent engineering and product lifecycle management. *Computers in Industry 64*(7), 833–848.

[8] Demoly, F., D. Monticolo, B. Eynard, L. Rivest, and S. Gomes (2010). Multiple viewpoint modelling framework enabling integrated product-process design. *International Journal on Interactive Design and Manufacturing 4*(4), 269–280.

[9] Demoly, F., X.-T. Yan, B. Eynard, S. Gomes, and D. Kiritsis (2012). Integrated product relationships management: a model to enable concurrent product design and assembly sequence planning. *Journal of Engineering Design 23*(7), 544–561.

[10] Fei, G., J. Gao, O. O. Owodunni, and X. Q. Tang (2010). A methodology for engineering design change management using modelling and problem solving techniques. In *Proceedings of the 36th International MATADOR Conference*, pp. 179–182. Springer.

[11] Fekete, J.-D., J. Van Wijk, J. Stasko, and C. North (2008). The value of information visualization. In *Information visualization*, pp. 1–18. Springer.

[12] Filos, E. (2012). The future of PLM. Experiences and Objectives of the European Research Programme.

[13] Fischer, M. and K. McKinney Liston (2001). 4d CAD Research - Wish list for 4d environments.

[14] Fonseca, F., M. Egenhofer, C. Davis Jr, and K. Borges (2000). Ontologies and knowledge sharing in urban GIS. *Computers, Environment and Urban systems* (24), 251–271.

[15] Fortineau, V., T. Paviot, and S. Lamouri (2013). Improving the interoperability of industrial information systems with description logic-based models - The state of the art. *Computers in Industry 64*(4), 363–375.

[16] Frank, A. (1997). Spatial ontology A geographical information point of view. Spatial and Temporal Reasoning. The Netherlands: Oliviero Stock.

[17] Gero, J. (1996). Design tools that learn: A possible CAD future. In *Information Processing in Civil and Structural Design, Civil-Comp Press*, pp. 17–22. Press.

[18] Gruhier, E., F. Demoly, O. Dutartre, S. Abboudi, and S. Gomes (2015). A formal ontology-based spatiotemporal mereotopology for integrated product design and assembly sequence planning. *Advanced Engineering Informatics 29*, 495–512.

[19] Gruhier, E., F. Demoly, K.-Y. Kim, S. Abboudi, and S. Gomes (2016). A theoretical framework for product relationships description over space and time in integrated design. *Journal of Engineering Design*.

[20] Hamilton, N. (2010). Managing change in manufacturing.

[21] Helms, R. W. (2002). *Product data management as enabler for concurrent engineering*. Ph. D. thesis, Technische Universiteit Eindhoven.

[22] Hoffmann, C. and R. Joan-Arinyo (2000). Distributed maintenance of multiple product views. *Computer-Aided Design 32*, 421–431.

[23] IBM Software (2012). The Rational Rhapsody family from IBM: Collaborative systems engineering and embedded software development.

[24] Issa, R., I. Flood, and W. O'Brien (2003). *4D CAD and visualization in construction Developments and Applications*. Taylor and Francis.

[25] Kiritsis, D. (2011). Closed-loop PLM for intelligent products in the era of the Internet of things. *Computer-Aided Design 43*(5), 479–501.

[26] Louhichi, B. and L. Rivest (2014). Maintaining consistency between CAD elements in collaborative design using association management and propagation. *Computers in Industry 65*, 124–135.

[27] Mignard, C. and C. Nicholle (2014). Merging BIM and GIS using ontologies application to urban facility management in ACTIe3d. *Computers in Industry 65*, 1276–1290.

[28] Ming, X. G., J. Q. Yan, X. Wang, S. Li, W. F. Lu, Q. Peng, and Y.-S. Ma (2008). Collaborative process planning and manufacturing in product lifecycle management. *Computers in Industry 59*, 154–166.

[29] Pan, Z., X. Wang, R. Teng, and X. Cao (2016). Computer aided design while engineering technology in top down modeling of mechanical product. *Computers in Industry 75*, 151–161.

[30] Parametric Technology Corporation (2006). Support PTC : Gestion des modifications et des configurations.

[31] Parametric Technology Corporation (2009). Manufacturing process management.

[32] Peachavanish, R., H. A. Karimi, B. Akinci, and F. Boukamp (2006). An ontological engineering approach for integrating CAD and GIS in support of infrastructure management. *Advanced Engineering Informatics 20*(1), 71–88.

[33] Pfoser, D. and N. Tryfona (1998). Requirements, definitions and notations for spatiotemporal application environments.

[34] Pikosz, P. and J. Malmqvist (1998). A comparative study of engineering change management in three Swedish engineering companies. In *Proceedings of the DETC98 ASME design engineering technical conference*, pp. 78–85.

[35] Pittet, P., C. Cruz, and C. Nicholle (2014). An ontology change management approach for facility management. *Computers in Industry 65*, 1301–1315.

[36] Pooyan, A., F. Griffis, and C. Lawrence (2009). Building information model the role and need of the constructors building a sustainable future. pp. 467–476.

[37] PTC (2010). Mastering change and configuration management for business advantage.

[38] Quintana, V., L. Rivest, R. Pellerin, and F. Kheddouci (2012). Re-engineering the engineering change management process for a drawing-less environment. *Computers in Industry 63*, 79–90.

[39] Rezgui, Y., S. Boddy, M. Wetherill, and G. Cooper (2011). Past present future of information and knowledge sharing in the construction industry towards semantic service based e construction. *Computer-Aided Design 43*, 502–215.

[40] Schlenoff, C., A. Pietromartire, Z. Kootbally, S. Balakirsky, and S. Foufou (2012). Ontology based state representations for intention recognition in human robot collaborative environments. USA, pp. 810–817.

[41] Sider, T. (2001). *Four dimensionalism: an ontology of persistence and time*. Oxford, Clarendon Press.

[42] Smith, B. (1996). Mereotopology: a theory of parts and boundaries. *Data and Knowledge Engineering 20*(3), 287–303.

[43] Smith, B. and D. Mark (1998). Ontology and geographic kinds. Canada.

[44] Smith, R. and S. Eppinger (1997). Identifying controlling features of engineering design iteration. *Management Science* (43), 276–293.

[45] Solutions, A. (2015). Three tips for creating a manufacturing change process that works.

[46] Stark, J. (2006). *Product Lifecycle Management. 21st century paradigm for product realization*. Berlin.

[47] Stell, J. G. and M. West (2004). A four-dimensionalist mereotopology. In *Formal Ontology in Information Systems*, pp. 261–272.

[48] Straume, K. (2014). The social construction of a land cover map and its implications for Geographical Information Systems (GIS) as a management tool. *Land Use Policy 39*, 44–53.

[49] Swain, A., D. Sen, and B. Gurumoorthy (2014). Extended liaison as an interface between product and process model in assembly. *Robotics and Computer-Integrated Manufacturing 30*(5), 527–545.

[50] Tavcar, J. and J. Duhovnik (2005). Engineering change management in individual and mass production. *Robotics and Computer-Integrated Manufacturing 21*, 205–215.

[51] Tremblay, T. (2006). *La propagation du changement dans un contexte PLM: vers la maitrise du role des associations dans le developpement de produits et des processus associes.* Ph. D. thesis, Ecole de Technologie Superieure.

[52] Weber, P. (2015). Organiser la gestion des modifications de conception - Technique de l'ingenieur.