



	A Product Lifecycle Management Framework to Support the Exchange of Prototyping and Testing Information.	
Auteur: Author:	r: Luc Boris Toche Fumchum	
Date:	2010	
Type:	Type: Mémoire ou thèse / Dissertation or Thesis	
<b>Référence:</b> Citation: Toche Fumchum, L. B. (2010). A Product Lifecycle Management Framework to Support the Exchange of Prototyping and Testing Information. [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. <a href="https://publications.polymtl.ca/436/">https://publications.polymtl.ca/436/</a>		

# Document en libre accès dans PolyPublie Open Access document in PolyPublie

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/436/
Directeurs de recherche: Advisors:	Clément Fortin, & François Provencher
<b>Programme:</b> Program:	Génie mécanique

### UNIVERSITÉ DE MONTRÉAL

# A PRODUCT LIFECYCLE MANAGEMENT FRAMEWORK TO SUPPORT THE EXCHANGE OF PROTOTYPING AND TESTING INFORMATION

# LUC BORIS TOCHE FUMCHUM DÉPARTEMENT DE GÉNIE MÉCANIQUE ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION

DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES

(GÉNIE MÉCANIQUE)

DÉCEMBRE 2010

## UNIVERSITÉ DE MONTRÉAL

## ÉCOLE POLYTECHNIQUE DE MONTRÉAL

#### Ce mémoire intitulé:

# A PRODUCT LIFECYCLE MANAGEMENT FRAMEWORK TO SUPPORT THE EXCHANGE OF PROTOTYPING AND TESTING INFORMATION

présenté par : TOCHE FUMCHUM Luc Boris

en vue de l'obtention du diplôme de : <u>Maîtrise ès sciences appliquées</u>

a été dûment accepté par le jury d'examen constitué de :

- M. VADEAN Aurelian, Doct., président
- M. FORTIN Clement, Ph.D., membre et directeur de recherche
- M. PROVENCHER Francois, M.Sc.A., membre et codirecteur de recherche
- M. PELLERIN Robert, Ph.D., membre

To Simo Anne and Fumchum Hubert.

#### **ACKNOWLEDGMENTS**

I wish to express my gratitude to Professor Clément Fortin for supervising this research and giving me the support and trust to carry out all the related activities.

I would like to thank Dr. Grégory Huet and Mr. Grant McSorley for their mentorship and friendship during these last couple of years.

I would also like to thank my industrial supervisor Mr. Francois Provencher from Pratt & Whitney Canada Corp., and Professor Vincent Thomson, the CRIAQ PLM2 research program investigator from McGill University, for providing me with appropriate context and environment for this research. Warm thanks also to the whole PLM2 group including the academic and industrial participants and the external visitors.

I acknowledge Professor Aurelian Vadean for presiding my thesis examination jury and Professor Robert Pellerin for being the internal examiner.

I must acknowledge the Mechanical Engineering Department and the whole personnel at École Polytechnique de Montréal for their kind support and advice. I'm also grateful to the 2009 and 2010 Virtual Environment Project students for their openness and participation. I equally thank all the members of the DEPPIA laboratory.

This work is the result of the research I have done as part of the *Collaborative Development for Product Life Cycle Management* project, CRIAQ PLM2 (mcgill.ca/plm2-criaq). I am grateful for the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC), the Consortium for Research and Innovation in Aerospace in Quebec (CRIAQ), Bombardier Inc., CAE Inc., CMC Electronics Inc., Pratt & Whitney Canada Corp. and Rolls-Royce Canada Limited.

Finally, I would like to thank my family and friends for their unconditional support.

### **RÉSUMÉ**

La conception et développement en tant qu'activité essentielle et motrice du cycle de vie du produit est désormais profondément influencée par les outils informatiques et leur utilisation par les divers acteurs, internes ou externes à l'entreprise et à travers les phases de vie du produit. Dans un contexte où chaque domaine d'expertise possède ses propres outils spécialisés, de nouveaux défis relatifs à la collaboration et à la dissémination de l'information sont présents. Ces défis vont crescendo aussi bien à cause de la croissance, la multiplicité et les relations entre les données générées que le souci de gérer leurs flux et des processus clés tels que le changement. A cet effet, les travaux présentés sont basés sur les transactions spécifiques d'un département de développement et tests dont les activités consistent à planifier et construire des prototypes afin de valider des performances fonctionnelles et opérationnelles du produit. Il s'agit donc de proposer et simuler une plate-forme s'appuyant sur les données configurées d'ingénierie, les structures d'information complémentaires et l'ensemble des pratiques modernes de gestion de données techniques pour converger vers des solutions et concepts répondant aux défis soulevés. Cette convergence devra en outre adhérer durablement à l'optique de gestion de cycle de vie du produit (PLM, Product Lifecycle Management). Dans un premier temps, une étude d'un cas de conception d'un nouveau pylône pour le remplacement d'un moteur de série d'avion est menée afin de détailler et implémenter, autant que possible, les fonctionnalités clés de la plate-forme cible. Cette simulation, mettant en exergue les structures complémentaires et configurables, est réalisée sur un système PLM récent à disposition pour le projet. Dans un second temps, une étude basée sur les logiciels libres et s'appuyant sur un modèle de communication préalablement justifié est déployée afin de démontrer comment des données nativement incompatibles peuvent être transformées, homogénéisées et gérées de manière consistante dans un emplacement commun. L'approche telle qu'implémentée en deux temps permet d'effectuer transpositions et extrapolations de manière à introduire l'Open Exchange Nest en tant que concept générique apte à supporter le travail collaboratif au sens PLM.

# CONDENSÉ EN FRANÇAIS

Le présent condensé en français vise à être suffisamment complet pour que la portée du mémoire soit saisie. Il conduira notamment à prendre connaissance du contexte et des objectifs de la recherche, discerner les étapes de l'activité de recherche, les analyses menées et leurs résultats et enfin, appréhender les principales applications ainsi que les conclusions et recommandations.

#### CONTEXTE ET OBJECTIF

L'industrie est aujourd'hui familière aux contraintes de plus en plus exigeantes que sont la complexité des produits, la nécessité de compétitivité passant par des temps de développement et mise en marché rapides, et plus ardu, l'établissement du contexte collaboratif supportant toutes les ramifications de l'entreprise et l'ensemble des activités prenant place au cours de la vie des produits. Il est dorénavant vital de disposer d'approches permettant aux multiples acteurs, œuvrant dans le cadre de l'ingénierie concourante, de détenir des informations adéquates aux moments opportuns afin de converger vers des optimums concertés plutôt que des optima locaux juxtaposés. La sécurité, l'intégrité ou l'échange d'informations entre intervenants étant autant de défis à relever en gestion collaborative.

La gestion du cycle de vie de produit (*PLM*, *Product Lifecycle Management*) se situe, en ce sens, à un niveau stratégique et consiste en un ensemble de solutions aussi bien en termes de systèmes d'information que d'organisation, processus ou méthodologies. La nécessité de gérer efficacement et de manière intégrée les informations définissant le produit demeure le cœur de l'approche et la gestion des données techniques en est la principale composante. Cette dernière est en effet une démarche permettant le contrôle de l'évolution d'un produit en fournissant les outils et procédures appropriés pour la distribution des informations précises sur ce produit. Ceci au bon acteur, dans le bon format et au bon moment durant le cycle de vie entier du produit.

Les travaux présentés dans ce mémoire de thèse s'inscrivent dans le cadre d'un programme de recherche unique en son genre intitulé «Développement collaboratif autour de la gestion de cycle de vie de produit». Ce programme, tel que synthétisé sur la figure 1.1, émane d'une initiative de l'université McGill et réunit cinq universités ou écoles et cinq entreprises majeures du secteur

aérospatial au Québec. Ces divers acteurs universitaires et industriels sont rassemblés autour de cinq thématiques orientées vers des solutions à fort potentiel opérationnel.

La troisième thématique porte sur la modélisation de processus. Elle vise à établir et simuler des transactions inhérentes au développement de produit de manière à refléter, d'une part, les pratiques adéquates pour cette phase critique du cycle de vie et, d'autre part, l'apport des fonctionnalités actuelles et futures des systèmes d'information dans l'optique d'une réelle efficacité de la démarche PLM. Cette thématique se décline en deux volets dont le premier traite des mécanismes de coordination en environnement collaboratif et le second, intérêt de la présente thèse, a trait aux méthodologies et caractéristiques des systèmes d'information permettant de garantir une véritable ingénierie simultanée. A cet effet, les travaux de recherche ici discutés sont centrés sur la collaboration et l'échange d'information entre deux entités de l'entreprise : l'une chargée de la conception du produit et l'autre du prototypage et des tests avant série. Avant de mieux préciser les objectifs et détailler la question de recherche, il est nécessaire de discuter quelques aspects importants concernant la structuration de l'information lors des phases de conception et développement.

#### ii. HYPOTHESES ET QUESTION DE RECHERCHE

Il est primordial de ne point perdre de vue, dans un domaine comme celui de l'aérospatial, la complexité des produits et des transactions, la multitude et la diversité des intervenants, la spécificité des données et des outils pour chacune des expertises ou plus encore le cloisonnement, typique de la gérance interne des entreprises de cette envergure. La réaction la plus instinctive dans ce type de contexte consiste à centraliser l'information ou en d'autres termes, implanter un modèle de donnée commun valable pour tous les intervenants et répondant à leurs besoins spécifiques dans ses diverses déclinaisons. Cette agrégation purement monolithique des données du produit lors de la phase de développement permet d'assurer consistance, intégrité, références communes et donc simultanéité et concrète collaboration. Ce choix conduit néanmoins à un échec car il devient rapidement très complexe de gérer le flux et la quantité de données générés par chaque intervenant. D'abord parce que l'ensemble des données et transactions d'une entité n'ont qu'une pertinence faible pour une autre, seuls certains éléments précis sont nécessaires à d'autres pour former un tout pertinent pour tous. Mais surtout parce que toutes les entités de

l'entreprise ne perçoivent pas du tout le produit de la même façon. L'équipe de conception voit des spécifications fonctionnelles et des composants s'imbriquant les uns aux autres pour fournir une performance requise. L'équipe de fabrication quant à elle voit des pièces manipulées sur des lignes d'assemblages, celle de prototypage et test perçoit des répliques approximatives et des instruments de mesure afin de rendre compte de certains niveaux de performance. Les équipes d'analyses par éléments finis, d'achats et sous-traitances, de certification, de qualité ou de relations clients, pour ne citer que celles là, ont toute chacune une vision propre du produit. D'où la dérive du modèle de donnée commun.

Compte tenu du fait que le produit virtuel, et la représentation dont il est possible de s'en faire jusqu'à sa réalisation découle de l'apport de toutes les entités impliquées dans le développement intégré, la notion de complémentarité de l'information est apte à constituer une solution viable dans le contexte mentionné plus haut. Par complémentarité il faut comprendre des structures d'informations dédiées à chaque entité et s'interconnectant de manière à consolider la chaîne de valeur tout au long du cycle de vie du produit.

Les maquettes numériques tridimensionnelles et les structures de produit configurées sont d'ores et déjà intensivement utilisées pour véhiculer l'information au sein de certaines entités, mais à prendre l'exemple de l'interaction entre les deux entités concernées dans ces travaux de recherche, les deux problématiques suivantes sont à considérer:

- Il n'existe actuellement aucune liaison tangible et stable entre la structure d'information de l'équipe de conception (structure d'ingénierie) et les structures de l'équipe de prototypage et test, c'est-à-dire celles correspondant aux tests physiques requis et aux procédures de test associées.
- Les processus liés au test physique et le retraçage de l'information concernant les prototypes construits sont des aspects qui ne sont pas pris en compte dans les systèmes de gestion de données techniques existants.

Partant de ces observations ainsi que de l'applicabilité de la notion de complémentarité de l'information, la question de recherche est articulée comme suit : Dans une perspective PLM, quelles sont les caractéristiques d'une plate forme apte à s'appuyer sur les structures d'ingénierie et les notions de gestion de configuration et de complémentarité pour permettre l'imbrication de l'information et des processus liés au prototypage et au test?

Il est à noter préalablement que la perspective PLM implique nécessairement une intégration de divers outils informatiques et comporte donc implicitement un problème d'interopérabilité dans le contexte de multi-partenariat ou simplement de cloisonnement.

La gestion de configuration, pour préciser ensuite, est une méthodologie ayant fait ses preuves dans l'armement et la défense. Celle-ci vise à implanter un contrôle strict des données du produit, dans chacune de ses variantes, afin de personnaliser la présentation pour chaque intervenant, mais aussi d'assurer l'intégrité de la documentation et de démontrer qu'elle correspond exactement au produit qui est livré, ceci pour des considérations aussi bien techniques, contractuelles que légales. La gestion de configuration est désormais largement appliquée dans toute l'industrie aérospatiale et fait partie des hypothèses de travail ici non seulement pour son omniprésence mais surtout pour son aptitude à relier dynamiquement une multitude de variantes à une représentation de base évolutive, comme ce sera nécessaire pour les prototypes et instances physiques.

Par ailleurs, l'applicabilité de la notion de complémentarité, tel que précédemment avancé, provient principalement des solutions qui existent déjà et qui sont opérationnelles. La pionnière, basée sur la gestion des processus de fabrication (MPM, Manufacturing Process Management) a été développée à l'École Polytechnique de Montréal. Elle réalise avec succès l'intégration de la Conception et des Méthodes à travers une interconnexion de la structure d'ingénierie et celle manufacturière. L'approche est implémentée dans un système PLM notoire qui sera en l'occurrence utilisé pour les simulations dans le présent travail de recherche. L'hypothèse centrale consistant à stipuler qu'eu égard aux similarités avec les processus de fabrication, les équipes de prototypage et test peuvent travailler sur les mêmes bases, avec une utilisation spécifique de certaines fonctionnalités, afin d'aboutir à l'intégration de leur information et le retraçage de leurs activités.

#### iii. ÉTAPES DE LA RECHERCHE

Les présents travaux de recherche suivent une boucle d'exploration présentée sur la figure 4.1. Cette exploration telle qu'organisée se calque aussi bien sur les livrables du programme de recherche pour le second volet de la thématique de modélisation de processus PLM, que la question de recherche qui en a été filtré. Les étapes du processus d'exploration dans leur ordre sont présentées ci-après :

- Caractérisation de l'activité multidisciplinaire et de l'infrastructure supportant sa mise en œuvre.
- Construction de cas d'études et scénarios basés sur les pratiques et technologies de l'information actuelles, ainsi que la vision PLM.
- Implémentation des scénarios dans un système PLM existant ; utilisation des structures d'information complémentaires et de la gestion de configuration.
- Prise en compte et traitement de l'absence d'interopérabilité afin de générer des concepts répondant aux besoins glanés à la deuxième.
- Synthétisation de la plate forme adéquate à partir des deux étapes précédentes, déduction des meilleures pratiques dans l'utilisation des outils dans la perspective PLM et conjectures sur les futurs systèmes PLM.

#### iv. ANALYSES, RÉSULTATS ET RECOMMANDATIONS

La plate forme cible a été construite suivant le mode exploratoire expliqué précédemment. Il s'agit d'une construction à deux volets, le premier traitant du problème d'interconnexion des structures d'information et, par ricochet, celui du retraçage des multiples instances physiques de prototypes. Le second volet s'attaque aux difficultés liées au cloisonnement et à l'échange de données entre outils nativement incompatibles.

#### a) Interconnexion des structures d'information

L'étude de cas est implémentée au sein du système PLM Windchill 9.1<sup>TM</sup> suivant les hypothèses indiquées dans la section ii. Cette étude de cas porte sur la modification des pylônes du CRJ-700 de Bombardier Aerospace afin d'y installer des moteurs PW305A de Pratt & Whitney Canada. Le scénario implique une équipe de développement effectuant cette modification d'ingénierie majeure et il inclut par conséquent l'ensemble des transactions ayant lieues dans un contexte de développement de produit intégré, d'utilisation intensive d'une maquette numérique tridimensionnelle et de gestion de données techniques configurées.

L'emphase est mise sur le support de fixation avant du moteur. Cette composante critique de la structure métallique du pylône réalise le lien entre le moteur et le fuselage. Elle répond aux principaux besoins que sont la fixation flexible du moteur et la transmission de la poussée au fuselage. La conception de la composante est dépendante des critères sévères habituels que sont la résistance mécanique, le poids ou encore les coûts de fabrication et de maintenance. Les prototypes sont construits et testés non seulement pour valider les performances requises mais aussi pour répondre aux requêtes des autorités de certification.

Partant de la structure d'ingénierie, l'équipe chargée du prototypage et du test s'appui sur la stratégie de fabrication et de test pour construire la structure complémentaire, support central pour leurs activités et pour le convoyage de leur expertise. L'interconnexion entre les structures parallèles est maintenue et mise à jour grâce aux liens d'équivalence, d'occurrence et de référence.

L'équipe de conception identifie des versions à tester au fur et à mesure du développement et crée donc des configurations qui seront remontées le moment venu. Celle de test agit idem et gère des configurations correspondantes au sein de la structure complémentaire. L'*instance* est déduite d'une configuration donnée et représente un prototype assemblé et testé. L'*instance* reflète bel et bien un prototype existant ou ayant existé parce que les pièces dans la structure, telle qu'arrangée, sont affectées des numéros de série ou de lot des pièces réelles. Sachant que plus d'un prototype peut être construit pour tester une configuration donnée, plusieurs *instances* y correspondantes peuvent coexister dans le système. De cette façon, des structures représentantes (de chaque prototype réel testé) et les informations spécifiques associées sont disponibles pour les deux équipes, celle de conception procédant également par l'instanciation tel qu'imagé sur la figure 4.6. De plus, dans le cas de l'équipe de prototypage et test, le banc de test, les instruments et tous les autres éléments ajoutés pour la réalisation du test sont aussi reflétés et retracés.

#### Cette analyse ainsi menée a permis de :

- Démontrer par quels moyens une liaison tangible et stable pouvait être établie entre la structure d'ingénierie et celle de prototypage et test, en supposant que cette dernière est équivalente sinon similaire à une structure complémentaire manufacturière.
- Démontrer que des processus liés au test physique et le retraçage de l'information concernant les prototypes construits pouvaient être implémentés sur la base de méthodes existantes, en l'occurrence la gestion de configuration.

La simulation réalisée dans Windchill 9.1<sup>TM</sup> est présentée en appendice.

#### b) Interopérabilité et échange de données

Pour les raisons déjà discutées plus haut, l'interopérabilité entre outils nativement incompatibles est un aspect à considérer lors de la définition d'une plate forme comme celle dont il est question ici. L'échange de données en environnement collaboratif est ainsi traité sur la base d'un modèle de communication amélioré. Ce modèle est construit en s'inspirant des théories générales de la communication et d'un modèle de transfert d'informations proposé par l'«institut national pour la technologie et la normalisation» (NIST, National Institute of Standards and Technology) aux Etats-Unis. Le modèle, tel que présenté sur la figure 3.4 est implémenté dans un scénario de collaboration afin de mieux cerner le type de situations auxquelles il se réfère, concrétiser l'ensemble des notions qui y sont présentées et en démontrer l'application. L'un des principaux éléments permettant la conversion et la transmission de l'information via l'interface de programmation du système informatique est le moteur de transformation. Celui utilisé dans la simulation est un logiciel libre provenant d'une initiative de l'École Centrale de Paris et dont l'École Polytechnique de Montréal participe à l'élaboration. Il met déjà en exergue la fédération de contenus, l'utilisation de langages, de schémas et de protocoles standards comme condition sine qua none à la réussite de l'échange d'information et la mise en commun des données.

Le concept de «nid d'échanges libres» (*OEN, Open Exchange Nest*) est ainsi introduit par synthétisation de l'ensemble des observations, analyses et résultats afin de fournir les caractéristiques de la plate forme cible telle que recherchée. La figure 4.7 présente deux déclinaisons du concept : la première étant la plate forme optimale correspondant au présent exercice d'échange entre un silo de conception et l'autre de prototypage et test et la deuxième une généralisation à l'interaction optimale entre divers silos tout au long de la vie du produit. Cette dernière déclinaison est en soi une recommandation pour le développement des futurs systèmes PLM.

#### **ABSTRACT**

The modern perspective on product life cycle and the rapid evolution of Information and Communication Technologies in general have opened a new era in product representation and product information sharing between participants, both inside and outside the enterprise and throughout the product life. In particular, the Product Development Process relies on crossfunctional activities involving different domains of expertise that each have their own dedicated tools. This has generated new challenges in terms of collaboration and dissemination of information at large between companies or even within the same organization. Within this context, the work reported herein focuses on a specific stakeholder within product development activities - the prototyping and testing department. Its business is typically related to the planning and building of prototypes in order to perform specific tests on the future product or one of its sub-assemblies. The research project aims at investigating an appropriate framework that leverages configured engineering product information, based on complementary information structures, to share and exchange prototyping and testing information in a Product Lifecycle Management (PLM) perspective. As a first step, a case study based on the retrofit of an aircraft engine is deployed to implement a scenario demonstrating the functionalities to be available within the intended framework. For this purpose, complementary and configurable structures are simulated within the project's PLM system. In a second step are considered the software interoperability issues that don't only affect Design - Testing interactions, but many other interfaces within either the company - due to the silo-arrangement - or the consortiums with partners, in which case the whole PLM platforms could simply be incompatible. A study based on an open source initiative and relying on an improved model of communication is described to show how two natively disparate PLM tools can dialogue to merge information in a central environment. The principles applied in both steps are therefore transposed to introduce the *Open* Exchange Nest as a generic PLM-driven and web-based concept to support the collaborative work in the aforementioned context.

# TABLE OF CONTENTS

DEDICATION	III
ACKNOWLEDGMENTS	IV
RÉSUMÉ	V
CONDENSÉ EN FRANÇAIS	VI
ABSTRACT	XIII
TABLE OF CONTENTS	XIV
LIST OF TABLES	XVI
LIST OF FIGURES	XVII
LIST OF ABBREVIATIONS	XIX
LIST OF APPENDIXES	XX
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 DIGITAL MOCK-UPS AND RELATED INFRASTRUCTURE FOR	
PROTOTYPING AND TESTING IN THE DEVELOPMENT PROCESS	6
2.1 Full virtual prototyping and limitations	7
2.2 Correlating design, simulations and physical tests	8
2.3 Digital mock-ups basis and usage	11
2.3.1 Geometry	14
2.3.2 Product structure and bill of material	15
2.3.3 Attributes	15
2.3.4 Configured DMU	16
2.3.5 Communication and management	17
2.4 PDM and MPM at the heart of the PLM infrastructure	19
CHAPTER 3 PROTOTYPING AND TESTING IN A LIFECYCLE MANAGEMENT	
PERSPECTIVE.	22
3.1 Prototypes in engineering	22

3.2 Phase-function view of the generic product development process	24
3.3 Cross-functional collaboration in heterogeneous environments	26
3.3.1 Models of communication	27
3.3.2 A scenario of collaboration in heterogeneous environments	28
CHAPTER 4 A TWO STEP FRAMEWORK FOR SHARING PROTOTYPING AND	
TESTING INFORMATION	33
4.1 The retrofit of an aircraft engine: A case study	34
4.2 Step one: Building and maintaining complementary and configurable structures	38
4.3 Step two: Facing software incompatibilities – the open exchange platforms	44
CHAPTER 5 CONCLUSIONS AND FUTURE WORK	47
LIST OF REFERENCES	50
APPENDIX	58
DISCLAIMER	79

# LIST OF TABLES

Table 1.1: Summary of the main tasks included in the research program	3
Table 2.1: Geometry data representation criteria	14
Table 2.2: Different application areas of physical and digital mock-ups (Dolezal, 2008)	18
Table 3.1: Properties of typical physical prototypes categories	23
Table 3.2: Key data for the collaboration scenario	29

# LIST OF FIGURES

Figure 1-1: PLM2 - Collaborative development for PLM	2
Figure 2-1: Moving from physical to virtual prototyping	7
Figure 2-2: Shift from Component-focused CAD/CAE/CAM to system-focused virt prototyping (Ryan, 1999)	
Figure 2-3: Combining physical test and simulation to deliver innovation (Van Der Auweraen Leuridan, 2005)	
Figure 2-4: Sustainable innovation potential driven by innovations in physical test and simulat (Van Der Auweraer & Leuridan, 2005)	
Figure 2-5: Digital Mock-up of a pylon to install a Pratt & Whitney PW305A engine on Bombardier CRJ-700 regional jet	
Figure 2-6: Three dimensions of DMU operation (Dolezal, 2008)	.13
Figure 2-7: Relationship between geometry and metadata in a simplified example	.16
Figure 2-8: Configured DMU attributes providing different product configurations	.17
Figure 2-9: DMU-centred collaborative development	.19
Figure 2-10: The six phases of the PDP and the 3 axes of integration in CE	.20
Figure 3-1: Three core qualities of a design artefact (Houde & Hill, 1997)	.22
Figure 3-2: Phase-function view of the generic product development process	.24
Figure 3-3: Experimentation as a four-step iterative cycle	.26
Figure 3-4: A model of communication in heterogeneous environments	.28
Figure 3-5: Mapping information to the central environment PLM schema with transformation engine Rialto Bridge	
Figure 3-6: Unified structure of the product and common and cross visualizations	.31
Figure 4-1: Steps for the design of the PLM framework supporting the identified transactions	.33

Figure 4-2: As-designed forward engine mount and corresponding 3D representation in the
context of the pylon's configured DMU
Figure 4-3: A Design arrangement of the forward engine mount components
Figure 4-4: Manufacturing complementary structure of the forward engine mount
Figure 4-5: Building and maintaining complementary structures: A user interface - Courtesy o PTC
Figure 4-6: Complementary and configurable information structures - a diagrammatic analysis .42
Figure 4-7: An Open Exchange Nest: application and extension

#### LIST OF ABBREVIATIONS

3D Three dimensions

API Application programming interface

BOM Bill of material

CAD Computer aided design

CAE Computer aided engineering

CAM Computer aided manufacturing

CE Concurrent engineering

CM Configuration management

DMU Digital mock-up

eBOM Engineering BOM

IPT Integrated product team

IT Information technology

mBOM Manufacturing BOM

MPM Manufacturing process management

OEN Open exchange nest

PDM Product data management

PDP Product development process

PLM Product lifecycle management

SOAP Simple object access protocol

STEP Standard for the exchange of product model data

XML Extensible mark-up language

WSDL Web services description language

# LIST OF APPENDIXES

APPENDIX A – Step one simulation in the PLM system Windchill 9.1TM [Design side]5
APPENDIX B - Step one simulation in the PLM system Windchill 9.1TM [Development &
Testing side]69

#### CHAPTER 1 INTRODUCTION

Recently, there have been important changes in the way engineered products are created, built, serviced and disposed of in order to meet both customer and market regulation requirements. This is not only characterized by the advances in computer support technologies (Waurzyniak, 2008, 2010) or the ongoing shift towards the virtual prototype that is marked by the spreading use of Digital Mock-Ups (DMU) (Lazzari & Raimondo, 2001). It is also related to the development process of those products (Ulrich & Eppinger, 2008), which entails critical principles and practices such as Concurrent Engineering (CE), cross-functional collaboration (Kim & Bum-Kyu, 2008), process parallelisation and integration (Fortin & Huet, 2007), Configuration Management (CM) and change management (Jarratt et al., 2005). These practices are industrial responses to the ever changing and competitive marketplace. As such, the amount and nature of information transactions among participants have changed in a remarkable manner and the common virtual representation of the product is now made up of many detailed and related data. The various stakeholders need to work concurrently across a secured network to provide all necessary information that covers, for example, physical prototyping and tests, customer satisfaction, environmental impact characterization or product disposal specification.

This product-centric information might be accessible from several locations and adapted for different domains of expertise during both running and future product development programs. It must continually provide the means for a simultaneous development process in the aim of reducing development costs and time, and improving in-service experience (McSorley et al., 2008). In this context, Product Lifecycle Management (PLM) is regarded as the ultimate supporting solution for product development (Grieves, 2005; Sääksvuori & Immonen, 2004; Stark, 2005) and is implemented by many companies to try to meet the aforementioned challenges.

Within such a context, valuable product information is scattered throughout various functional areas in the company, and the PLM challenge is therefore to provide information and process driven approaches to implement an integrated cooperative and collaborative management of *product data*, throughout the entire product life (Fortin & Huet, 2007; Liu et al., 2009). The research reported herein is part of a wider project entitled *PLM2 – Collaborative development for Product Lifecycle Management*.

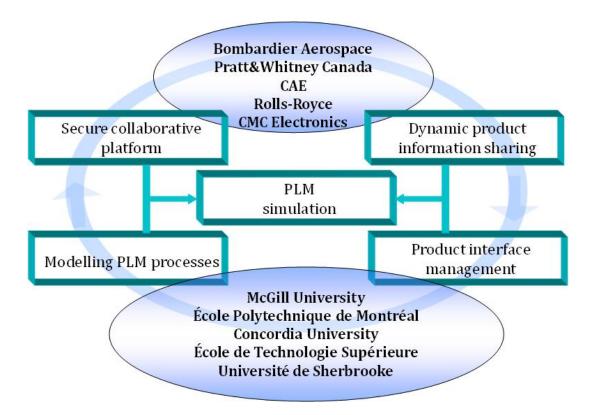


Figure 1-1: PLM2 - Collaborative development for PLM

As depicted in figure 1.1, the project is an innovative partnership between five universities and five major aerospace companies based in the province of Quebec, Canada. Participants are exploring current and future collaborative environments to enhance the effect of information systems functionality on PLM performance and make new product development better, faster and cheaper (PLM2, 2010). The research program is divided into five tasks briefly detailed in table 1.1. Since all the companies engaged in the program are in the process of implementing or expanding their use of PLM systems, the research environment provides a wealth of relevant information on aerospace product development and lifecycle management in general. Regarding the task *Modelling PLM processes*, extensive discussions with industry have corroborated the aforementioned main challenges concerning collaboration and dissemination of information in cross-functional teams work.

Table 1.1: Summary of the main tasks included in the research program

Task	Description	Key participants
Dynamic product information sharing	The development of a protocol for data sharing, that will act as a 'docking station' for Computer Aided Design (CAD) models to support the removal of CAD files from and their reintroduction into information systems.	Rolls-Royce & École de Technologie Supérieure
Secure collaborative platform	The development of a collaborative design methodology that can accommodate the interactive exchange of ideas, models and designs within a secure environment.	Pratt & Whitney Canada & Concordia University
Modelling PLM processes	The creation, modelling and simulation of New Product Development (NPD) processes that reflect best practices and best functionalities in IT tool design and tool use for PLM.	Bombardier Aerospace, Pratt & Whitney Canada, McGill University & École Polytechnique de Montréal
Product interface management	The development of a methodology for defining the function and requirements of system interfaces (mechanical, electrical, hydraulic, software, data exchange, control, etc.) for better sharing of product information.	CAE, McGill University & Université de Sherbrooke
PLM demonstrations	The modelling of PLM best practices and interface management techniques into a demonstration of an improved PLM process.	All participants

It has also been observed from previous research in the field that the monolithic product representation is inadequate to support the necessary transactions for an effective cross-functional collaboration during the product development (Brissaud & Tichkiewitch, 2000; Fortin & Huet, 2007; Szykman et al., 2001; Zimmermann et al., 2002); being it internal to the company or shared with some partners. Indeed, when it comes to the testing and refinement phase as part of the Product Development Process (PDP) the following concerns have been raised:

- The required physical tests and procedures and the derived *as-built* structures are not systematically connected to the *as-designed* structure.
- Hardware testing transactions and prototype information tracking are not addressed within common PLM systems.

From these prospects, the main arising question is: what is an appropriate framework that leverages configured engineering product information, based on complementary information structures, to support the management of prototyping and testing information in a PLM perspective?

The goal of the work project reported herein is therefore to provide a framework that capitalizes on DMU configurations, complementary information structures and the development & testing information specificity to support the needed exchange of prototyping and testing information. This research is, indeed, deployed following the objectives and deliverables defined for the CRIAQ PLM2 subtask *IT tools for PLM* included in the main task *Modelling PLM processes*:

- Define best functionality in IT tool design and best tool use for PLM.

The aim being to explore new functionalities needed to best support PLM processes and raise issues concerning their software implementation in PLM approaches.

- Simulate PLM processes showing the effect of IT tool functionality.

As a test environment, Windchill 9.1<sup>TM</sup>, a PLM tool from Parametric Technology Corporation (PTC), presents capabilities in handling critical principles and practices such as CE, cross-functional collaboration, process parallelisation and integration, CM and change management. It is therefore proposed to simulate a number of predefined scenarios using a Digital Mock-Up and data from the virtual environment student project which takes place at École Polytechnique de Montréal.

- Describe the impact on future IT system design given different levels of desired information system features.

The results from simulating the scenarios and the meetings with some active PLM vendors should provide new challenges in IT system design and show the potential impact of the research on future PLM systems. New concepts and approaches to support collaborative developments and the sharing and exchange of information in a PLM perspective are therefore to arise.

The thesis is subdivided into five chapters, the present introduction being the first. The next serves primarily to characterise information in the context of PLM by discussing some critical concepts regarding product data and DMUs within the PDP. An insight into virtual prototypes and DMUs usage is first provided. The correlation between simulation and physical tests is then

discussed to justify the necessity to start organizing the testing information so as to be linked to *as-designed* structures and to mirror the built prototypes within the evolving PLM infrastructures.

The role and place of the testing and refinement phase during the development process is discussed in the third chapter. Key aspects concerning prototyping and testing transactions are also highlighted. Since the main challenge lies in managing interfaces with other teams, information sharing and exchange are subsequently discussed on the basis of a model of communication suitable for facilitating collaboration in silo-arranged contexts at large. A collaboration scenario is then presented illustrating how two natively disparate tools can dialogue to merge information in a central environment while adhering to modern PLM requirements. This scenario serves to both illustrate the proposed communication model and highlight the necessary elements enabling a successful interoperability. Furthermore, when solely tackling the interconnection between structures, one can observe that some PLM platforms already operate Design - Manufacturing integration via, principally, the engineering and manufacturing BOMs. The hypothesis stipulating that the Design – Testing interaction can be supported as well or to a certain extent with such platforms is then formulated.

In chapter four, a case study based on the retrofit of an aircraft engine is therefore deployed to explore the preceding hypothesis and to implement a scenario demonstrating the functionalities to be available within the intended PLM framework. For this purpose, complementary and configurable structures are simulated within the project's PLM system, Windchill 9.1<sup>TM</sup>.

The outcome from the simulation in the available system and the observations made within the collaboration scenario in Chapter 3 are transposed to compose a two step framework. As it copes with generic issues in the aerospace industry, particularly in the second step, the introduced framework should also be seen as an extrapolation from present PLM systems towards those of the future.

In chapter five, the conclusions arising from this work are presented along with the intended future work.

# CHAPTER 2 DIGITAL MOCK-UPS AND RELATED INFRASTRUCTURE FOR PROTOTYPING AND TESTING IN THE DEVELOPMENT PROCESS

Economists have recognized the value and difference of information as compared to other resources (Warsh, 1998): the future of a company is almost completely driven by its ability to render its products or services in a digital form (Negroponte, 1995). The vision of replacing atoms by bits is becoming true. This necessitates a vision of enabling a correspondence between digital representations and physical components, as well as their surrounding environment and related lifecycle processes. From that viewpoint, the product design, fabrication, usage and even disposal can be assessed earlier and whenever for thorough optimisation of both the product and the processes. The term digital enterprise technology has been coined by Maropoulos to define the collection of systems and methods for the digital modelling of the global product development and realization process, in the context of lifecycle management (Maropoulos, 2003). The approach is still hard to implement but is strongly being undertaken. Tools to capture and simulate the physical product and the related instances and behaviours are firmly evolving. The trend is toward the concept of true virtual product development (Waurzyniak, 2008) which reflects the ability, through the use of a computer, to design, manufacture, test, and even service products virtually, before physical design or tooling construction is really started. Figure 2.1 translates this ultimate vision. The upper part of the figure displays the current, conventional way of moving from a concept to a performing artefact. The lower part, to the other extreme, discloses the digital definitions, transactions and assessments as a whole, which is then the only intermediate state between the concept generation and the production, the service operation included.

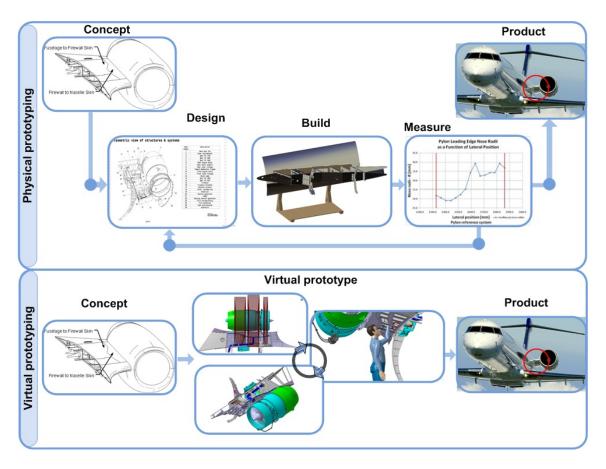


Figure 2-1: Moving from physical to virtual prototyping

# 2.1 Full virtual prototyping and limitations

The virtual prototype is presented and understood here as an integration of data from various sources to define the total product and its environments; It provides superior means of visualizing any aspect of the product design, its fabrication and assembly and the environment it will be used in (Coyle & Paul, 1997). Even if reality is not yet so close to this ideal, virtual product development has nevertheless been initiated with the introduction of three dimensional (3D) modelling within Computer Aided Design (CAD) systems. Related tools such as Computer Aided Engineering (CAE) and Computer Aided Manufacturing (CAM) have considerably improved part design and manufacture (McMahon & Browne, 1993). Now, the Digital Mock-Up (DMU) is paving the way for system-level design, simulation and test; Figure 2.2 illustrates this envisioned shift. The DMU is to assess the form and fit of assemblies of 3D solid models constituting the product. Functional virtual prototyping aims to assess the operating function of the assembled

product and the virtual factory simulation is to investigate the manufacturing and assembly of the product (Lazzari & Raimondo, 2001; Ryan, 1999).

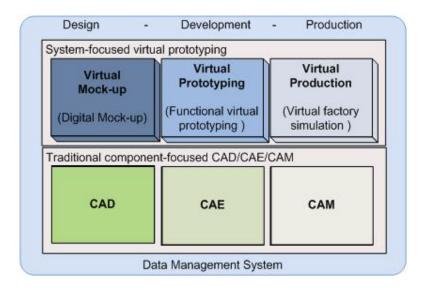


Figure 2-2: Shift from Component-focused CAD/CAE/CAM to system-focused virtual prototyping (Ryan, 1999)

Authors argue that the combination of the digital mock-up, the functional virtual prototyping and the virtual factory simulation provides means to move from hardware prototyping practices to software prototyping and therefore eliminates the expensive prototypes that have to be built to verify the product functions and behaviour. However, some crucial limitations, chiefly in functional virtual prototyping, are still present today:

- Lack of technology to accurately represent the components' behaviour and their cross functional relationships under variable situations.
- Indispensable role of hardware prototypes in manufacturing organisation usages.
- Roadblocks to acceptance of process change, training and adoption.

# 2.2 Correlating design, simulations and physical tests

Since the former limitations remain true, the aim of enhancing the use of virtual environments during the development should rather be toward a synergetic use of both physical and digital representations than the radical elimination of the first. In fact, hardware testing is done with more confidence and can sometimes truly be executed more quickly and at a lower cost for

tangible outcomes and rapid feedback promoting learning (Carleton & Cockayne, 2009). Van Der Auweraer et al. strengthen the view of mixing physical tests and simulations by indicating that physical test methods should be used to validate and calibrate simulation models and thereby extend the applicability of these last ones. Figure 2.3 shows how combining physical tests and simulations delivers innovation (Van Der Auweraer & Leuridan, 2005). The Y-axis represents the required capability for some engineering tasks, specifically system verification, and the X-axis represents the overall effort needed to accomplish the tasks.

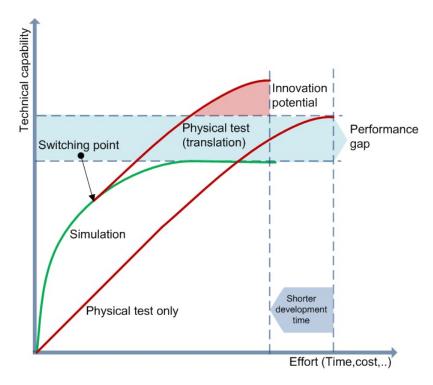


Figure 2-3: Combining physical test and simulation to deliver innovation (Van Der Auweraer & Leuridan, 2005)

With simulations only, the available technical capability can be used very fast and the development time is then shortened. With physical tests solely, much more effort is needed to use the greater available engineering capacity at its maximum, but uncertainties are firmly eliminated at each attempt (Gerber, 2009). Switching from simulations to physical tests by exploiting the simulation's static results considerably reduces the effort to benefit from the overall technical capability. It also opens new fields for the validation and exploration of the product behaviour and therefore for innovation. Hence, any novelty either in physical test or simulation increases the engineering capability, extends the exploration potential and consequently exposes latent

possibilities in product innovation. Figure 2.4 illustrates this sustainable innovation driven by new developments in physical tests and simulations. The optimal combination of simulation and physical test not only provides better system performance exploration, refinement and certification but continually reduces development time, strengthens virtual prototyping and finally opens new solution spaces.

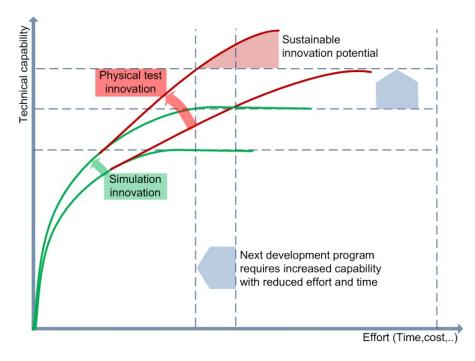


Figure 2-4: Sustainable innovation potential driven by innovations in physical test and simulation (Van Der Auweraer & Leuridan, 2005)

However, the simulation-to-test switching point could be difficult to determine in practice basically because the current virtual product representations mainly deal with geometry and materials and they are strongly oriented to the way to produce the product rather than the way it behaves in its physical environment. This is typically due to the lack in application of behavioural oriented descriptions of components to assess system functions under diverse circumstances, even when considering the advances made in Multidisciplinary Design Optimization (MDO) methods (Panchenko et al., 2002). Indeed, correlation between design activities and virtual and real system testing is not established and the implications of the testing results on the corresponding CAD/CAE-simulation models have rarely been addressed up to now (Riel & Brenner, 2004). It is therefore necessary, as a first step, to provide the means of linking these specific types of information and transactions within the existing PLM contexts. In addition, any attempt in doing

so and, furthermore, leverage physical test and simulation combination for whole product refinement should stem from current industry practices and paradigms and internal information technology infrastructures.

The notion of *product data* and a new perspective on the DMU, as drawn from the literature, are therefore presented in the next section to grasp the basis on which information is currently disseminated throughout the company. The critical role the DMU plays in the modern development process, how it is used in prototyping and testing activities and the reciprocal impact on physical product realisation are discussed. This will justify the necessity to start organizing the testing information so as to be linked to *as-designed* structures and mirror the built prototypes.

#### 2.3 Digital mock-ups basis and usage

The influence of 3D modelling and its central role in traditional product simulation and development have been observed in section 2.1. CAD systems and fast evolving Information Technology (IT) in general have opened new horizons in product representation and product information sharing between participants; inside and outside the enterprise and throughout the product life. As a matter of fact, product complexity combined with geographic dispersion, domain expertise and tools specialization, have raised concerns in collaboration and dissemination of information.

Within such a context, valuable product information is scattered throughout various functional areas in the company, and the PLM challenge is therefore to provide an information and process driven approach to implement an integrated cooperative and collaborative management of *product data*, throughout the entire product life (Fortin & Huet, 2007; Liu, et al., 2009).

From this prospect, the *product data* and the *product data model* are some key concepts to consider; the *product data*, which basically refers to all informational entities related to the product, can be clustered into three types (Sääksvuori & Immonen, 2004):

The specification data technically describe the physical, logical and functional properties of
the product. It is through these descriptions that stakeholders transmit their expertise.
Sketches, CAD models, drawings, FEA, NC files, Test Plan files are well known examples of
documents that contain large amounts of this type of data.

- *The metadata* is the so called "data about data" that serves to locate, identify, trace, retrieve and eventually describe it for an adequate use of its embedded knowledge.
- *The lifecycle data* identifies the status, maturity and progression of product information. The lifecycle data is a specific sort of metadata and is useful for the flow and processing of product information throughout its lifecycle.

The *product data model* reflects the conceptual representation of the product that serves to enclose and to deploy the various connected information elements and objects. The Product Structure (PS) or Bill of Materials (BOM), as perceived from a specific viewpoint, is frequently used as the main data source for the *product data model*. However, the DMU is also seen as a clearly defined set of data in the *product data model* (Döllner et al., 2000) and, as detailed below, several definitions can be found in the literature:

- For Gausemeier et al., the DMU and the virtual prototype are similar: it is the basic idea to create computer models for all relevant aspects of the product in development and to analyse them. Thereby can the time and cost consuming constructions of real prototypes be reduced (Gausemeier et al., 2000).
- For Berchtold, the DMU is an intrinsic compound of the development context and even establishes it: it is a complete virtual working environment for the whole process chain of 3D development and support of complex products with integrated effectivity and variant control (Berchtold, 2000).
- Dolezal defines the DMU as a digital 3D representation of a product together with its product structures and attributes (Dolezal, 2008). Figure 2.5 illustrates this view.

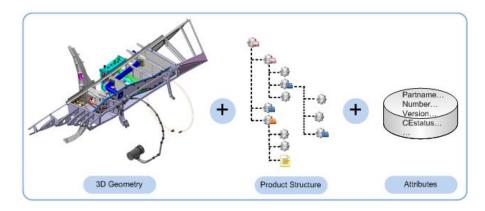


Figure 2-5: Digital Mock-up of a pylon to install a Pratt & Whitney PW305A engine on the Bombardier CRJ-700 regional jet

The latter definition is retained here because it rather describes what it is than what it does or how it can be used and thus reduces confusion due to the vast fields of application DMUs have. Figure 2.6 exhibits the various aspects of the DMU and the potentials and benefits of its use. Indeed, the DMU can be considered as fulfilling a core supporting role for three typical dimensions of interest in PDP transactions, namely for technical, communication, and management purposes.

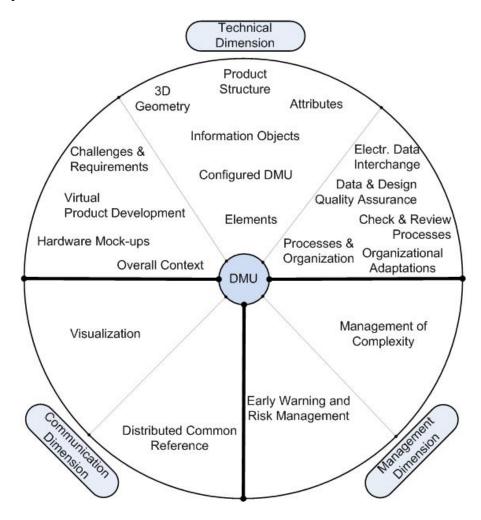


Figure 2-6: Three dimensions of DMU operation (Dolezal, 2008)

Such processes as *Data & Design Quality Assurance* are part of the technical dimension of DMU operation as shown in figure 2.6. This means that the DMU is used as a core reference for these processes and not that the DMU systematically includes those processes' data. Regarding the technical dimension, and to complement this insight into the DMU basis and usage, geometry

and product structures are discussed below as the main ways of rendering and using the classified data for an effective implementation of embodiment, prototyping and manufacturing activities. Also, the notions of attributes and configured DMU are detailed as they significantly contribute to data presentation, management and traceability throughout all development activities.

#### 2.3.1 Geometry

As basic specification data, 3D models provide an insight into a component's shape, functions, and furthermore into design intents. While not necessarily representing manufacturing tolerances or operational deformations, 3D models are the closest digital replicas of the parts to be produced. Therefore, they serve as the main three-dimensional visual references for all participants. The level of detail of the models depends both on the lifecycle stage and the objectives and requirements of the ongoing activity. As shown in table 2.1, the geometry can be managed under different formats either being lightweight approximations or exact models with the data volume reduced or not (Dolezal, 2008).

Table 2.1: Geometry data representation criteria

Criteria	Representation (stored in database)
Exact, not data volume reduced	CAD native, Constructive Solid Geometry
Exact, data volume reduced	Boundary Representation
Approximated, data volume reduced	Tessellation, Voxel, Texel, Octree, etc.

Geometry represents a critical element in PDP when it comes to product simulation and test. Several examples to enhance geometry use and level of granularity across dependent transactions can be found in literature: Feature-based modelling and feature recognition techniques for intelligent CAD (Cuillere et al., 1997; McMahon & Browne, 1993; Shah & Mäntylä, 1995); Shapes adaptation processes, from CAD reference representations, to produce appropriate geometric shapes and correct scene semantics for the targeted scenarios (Drieux et al., 2006); Graph-numerical parameters to allow the integration of information coming from different trade practices in a DMU (Danesi et al., 2008) or Key Characteristics (KCs) methodologies to a large extent (Zheng et al., 2008).

#### 2.3.2 Product structure and bill of material

The Bill Of Material (BOM) represents a particular way of aggregating and presenting product data by disclosing hierarchical and logical dependencies among parts and all relevant attached objects. The BOM, effectively a product structure perceived from a specific viewpoint, serves to organize the *product data* in such a way that facilitates access to it by considering the appropriate field of expertise of the user and the product's lifecycle maturity. Two BOMs are frequently encountered:

- The *as-designed* view which is typically a functional decomposition of the product, and thus discloses its systems arrangement with interface control objects. It is the engineering BOM (eBOM). The product, sub-assembly and component design data are accessible through this type of structure.
- The *as-planned* view refers to a structure reflecting how the product is to be manufactured and assembled from a process planning perspective. It is the manufacturing BOM (mBOM). The manufacturing resources and process plan data are accessible through this type of structure.

Depending on the lifecycle approach applied in the company, many other views could be found such as maintenance BOM and quality BOM (Brissaud & Tichkiewitch, 2001).

In particular, prototyping and testing activities are to end up with *as-built* structures that identify and mirror physically assembled prototypes. Some additional components and relevant instrumentation to assess the prototype's performances are included. The manufacturing strategy of the prototype and the specific test procedure both fashion the *as-built* structure. The same pattern as for designing the final product mBOMs should be followed to generate these *as-built* structures as it will be discussed in chapter four.

#### 2.3.3 Attributes

Attributes include all the metadata and lifecycle data for effective information management and distribution. Not restricted to information identification, attributes are key enablers for traceability and concurrent work. The relationship between the three main elements of the DMU is shown in figure 2.7.

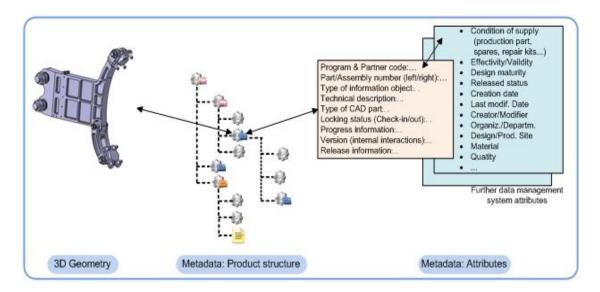


Figure 2-7: Relationship between geometry and metadata in a simplified example

The product structure is here presented also as a type of metadata because it is not limited to the display of a stored file. It is rather the result of a dynamic construction that uses on one hand, the links between items and on the other, the viewing criteria defined by the user's context and the status of attributes.

### 2.3.4 Configured DMU

A configured Digital Mock-Up combines Configuration Management (CM) and 3D design. CM is a management process ensuring that (ANSI/EIA-649-A, 2004):

- Products conform to the design and documentation governing their development and production.
- Documentation is controlled and reflects the latest, approved version.
- End users will have the capability to maintain or re-obtain the same delivered components.

To achieve these objectives, the aim of a configured DMU during the development process is to control and provide the digital product representations corresponding to each variant so as to be manipulated by the multitude of engineers who work on them simultaneously, regardless of the lifecycle stage. As an example, figure 2.8 shows how product variants stem from DMU elements controlled by a few configuration attributes (Dolezal, 2008). During the development, configuration involves the control of iterations on the product data from design, manufacturing or any other stakeholder working via the DMU. By relying on numbering, ranking and serialising

schemas, an advanced effectivity management engine is used to generate up-to-date representations that match each filter or request.

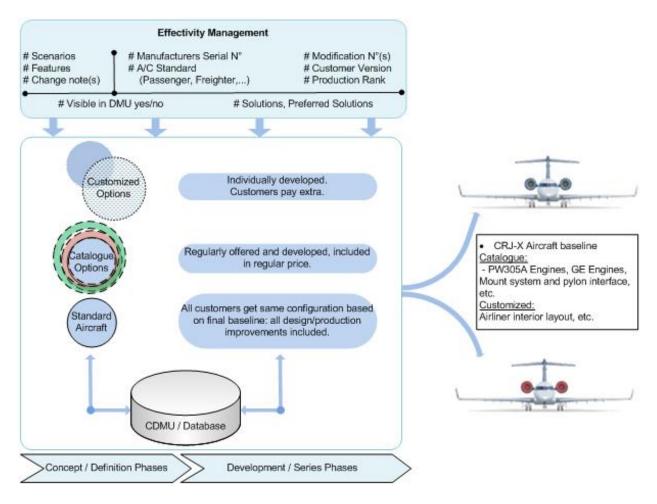


Figure 2-8: Configured DMU attributes providing different product configurations

The configured DMU is of a particular importance in this study because it is through it that the diverse prototypes and test cases are represented in parallel in Chapter 4. The inferred physical instances are then mirrored and tracked for further analysis and design refinement.

## 2.3.5 Communication and management

As compared to physical mock-ups, the DMU's communication and management dimensions presented in figure 2.6 are to enable:

- For the former, the visualization of a distributed common reference which is necessary to succeed in cross-functional activities.

- For the latter, the management of complexity via a dispatching of early warnings and a global risk management. Complexity being understood here as any qualitative or quantitative aspect of the project which is not wholly deterministic with conventional methods.

As such, a great part of prototyping and testing objectives is attained when the DMU is extensively used during the development process. The remaining is met, of course, when the physical prototype is built and is performing as intended, with supplemental modifications or not. To illustrate, table 2.2 shows different application areas of physical mock-ups and digital mock-ups distinguishing primary and secondary focuses.

Table 2.2: Different application areas of physical and digital mock-ups (Dolezal, 2008)

	Design/Prototyping/ Manufacturing	Training	Marketing
Primarily used	Digital MU  To represent the product itself but also for its entire production means (factories, transportation equipment) and verification of servicing procedures.	Physical MU  Mock-ups for workers being assigned to a new assembly line for example e.g.: Space Shuttle Training Mock-up, Fuselage/Cabin	Physical MU  To provide customers with a "look and feel" experience, e.g. with fully functioning components – the "Sales Mock-up"; (scaled) Mock-ups for exhibitions"
Secondarily used	Physical MU To validate particular risk areas, to cover certification relevant items, to prove required functions (system tests) that are not yet reliably possible in a digital environment.	Digital MU  To support faster and better learning e.g.: Space Mission training"	Digital MU  For external communication especially when coupled with Virtual Reality techniques; increased reactivity on customer needs and requirements.

Along with enabling both data management and risk management, digital mock-ups are also to mirror physically assembled, tested and serviced products to enable an efficient feedback loop to everyone involved in the refinement. This is only possible with an adequate infrastructure which is currently approached with Product Data Management (PDM), Manufacturing Process

Management (MPM) and Product Lifecycle Management (PLM) systems to a large extent. These systems are presented in the next section to clarify the as-is context and frameworks deployed in industry to meet the requirements of collaborative product simulation and realisation.

#### 2.4 PDM and MPM at the heart of the PLM infrastructure

PDM is an essential enabler for PLM (Stark, 2005) since it includes key functionalities related to the virtual product such as data vault and document management, structure and configuration management, data sharing and exchange, pre-visualization and notifications. PDM also supports key activities such as approvals or engineering change processes. Common PDM implementations are built on the item concept, which refers to an informational package relevant to transactions when populating a DMU. As such, items encompass both the information and the definition of data flow (Rangan et al., 2005). Parts and documents in PDM systems, such as CAD models, pdf documents or MS Excel spread sheets, are some illustrations of items which embed the form, the fit and eventually the function of the constituents of the product. All this is supported by a framework using the DMU as the common representation of the intended physical instances. Figure 2.9, in contrast with figure 2.1, illustrates this DMU-centred collaborative work as drawn from current industry practices.

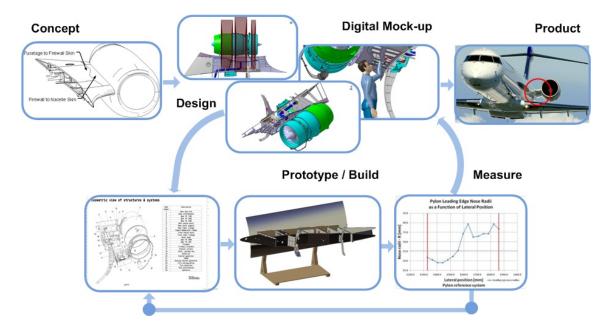


Figure 2-9: DMU-centred collaborative development

As far as manufacturing is concerned, the MPM platform helps to bridge the distinct worlds of engineering and production by focusing on the manufacturing process definition of the product (Huet et al., 2009). It takes advantage of the complementary information structures, via the mBOM, and full CAD representations to converge towards an optimal parallelisation of design and manufacturing processes through the digital collaborative environment. Figure 2.10 exhibits the central role that MPM plays in the Concurrent Engineering (CE) integration paradigm (Fortin & Huet, 2007). The figure highlights the information flow requirements between the main supporting tools for each key product development domain. As displayed, the integration of Supply Chain Management (SCM), Customer Relationship Management (CRM) and Enterprise Resources Planning (ERP) is equally sought in the PLM approach.

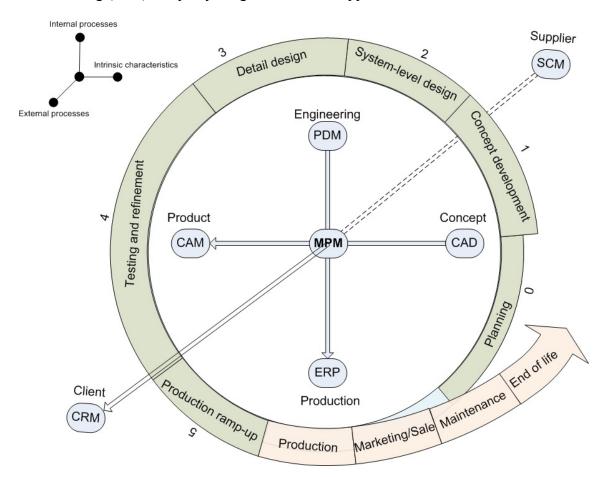


Figure 2-10: The six phases of the PDP and the 3 axes of integration in CE

Based on figure 2.10, the focus of the study reported here is the concept-to-product axis, which includes the engineering and prototyping transactions implemented to develop a single product. This axis emphasizes the spatial and physical embodiment of the product. Its core

supporting tools are CAD and CAM for respectively 3D simulation and assistance in fabrication. MPM is central to the axis since it has a significant role in developing the manufacturing strategy, based on the adequate data presentation, linkage and management.

Some dedicated tools for prototyping and testing activities are found in industry. However, even if they retrieve engineering product information in a transactional mode, they are not explicitly part of an integrated value stream and/or loop (Toche et al., 2010). The result is a situation where the development and test information system helps manage the construction and test of multiple prototypes but all relevant information pertaining to them remain scattered for upstream and downstream visibility. This is typically due to a lack in data structures and interconnections, on one hand, and the state-of-the-art regarding DMU-supported processes on the other.

This first insight into information and systems supporting the PDP throughout its main stages illustrates the increasing role that IT now plays in the life of a product. It also discloses the necessity to start organizing prototyping and testing information in accordance with the ongoing changes in engineering practices as generated by the PLM vision. The next chapter discusses the role and place of the testing and refinement phase during the PDP. It also presents key aspects concerning the implementation of prototyping and testing activities in a PLM perspective.

# CHAPTER 3 PROTOTYPING AND TESTING IN A LIFECYCLE MANAGEMENT PERSPECTIVE

PLM has previously been defined as an information and process driven approach to implement an integrated cooperative and collaborative management of *product data*, throughout the entire product life (Fortin & Huet, 2007; Liu, et al., 2009). It is for this purpose that the main objective of the work reported here pertains to capturing and representing the prototyping and testing transactions to effectively enable a lifecycle management context.

#### 3.1 Prototypes in engineering

Prototypes are generally built to examine design problems and evaluate and refine solutions. If one considers an artefact to be an end-result of a design activity, a prototype can be said to be built with the purpose of measuring one, two or the three core qualities of the artefact, namely its role, the implementation and the look and feel (Houde & Hill, 1997). These qualities can be defined as follows:

- The role refers to the function and how it corresponds to the user's need.
- The implementation refers to the constituent parts and logic through which the function is performed.
- The look and feel is about the sensory experience of the user.

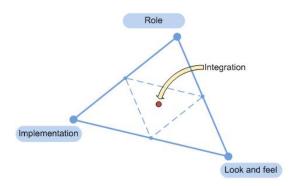


Figure 3-1: Three core qualities of a design artefact (Houde & Hill, 1997)

A prototype can therefore be defined as an approximation of the product along one or more dimensions of interest (Ulrich & Eppinger, 2008). Besides the essential purpose of learning by experimentation, prototypes are used during the PDP for three other main objectives:

communication, systems integration and milestones (Lazzari & Raimondo, 2001; Ulrich & Eppinger, 2008). Prototypes are usually classified according to the degree to which they approach reality and, as discussed in chapter one, a differentiation is made between the digital prototypes (e.g. DMUs) and the physical ones which are necessary to detect unanticipated phenomena (Gerber, 2009). During the conventional development process, the prototypes are generally regarded as the first physical expressions of the concept and are usually built without using mass production infrastructure or tooling (Clark & Fujimoto, 1991). To illustrate, table 3.1 presents the three categories, namely alpha, beta and preproduction, under which fall the prototypes typically built during the PDP. These categories are defined by the prototype's main objectives, as well as its similarities and differences with respect to the production version.

Table 3.1: Properties of typical physical prototypes categories

	Main objectives	Similarities	Differences
Alpha prototypes	Assess whether the product works as intended	Geometry, material	Production processes, suppliers
Beta prototypes	Assess reliability and identify remaining bugs in the product;  Test in the intended use environment (by customers).	Geometry, material, production processes, suppliers	Assembly facilities and tooling
Preproduction prototypes	Verify production process capability First supplies to preferred customers	Geometry, material, production and assembly processes, suppliers	Full capacity production facilities

The listed objectives deal primarily with the concept and process performance validation. These include all the testing activities preceding the serial production. It must however be noted that prototyping and testing activities are also carried out in the aerospace industry for certification issues, mature technologies introduction as well as for investigation of failures in the field. As such, prototyping and testing are done, for the great part, during the PDP but are not excluded from appearing well earlier or later in the product lifecycle.

#### 3.2 Phase-function view of the generic product development process

Figure 3.2 exhibits the generic PDP with its six phases and includes the tasks and responsibilities of the key functions of the organization for each phase (Ulrich & Eppinger, 2008). Key activities in the scope of the present research are highlighted.

The product development process could be defined as the different design, engineering, and manufacturing processes involved from the definition of the market needs to the end of the production ramp-up (the point in time when the satisfactory manufacturability of the product is reached) (Fortin & Huet, 2007). The authors approach product development from a systematic perspective and address concerns on how to carry out the process in the current information-scattered environment that characterises the modern enterprise. They therefore argue that the use of complementary information structures and an optimal parallelisation of the processes can significantly improve the PDP through the digital collaborative environment.

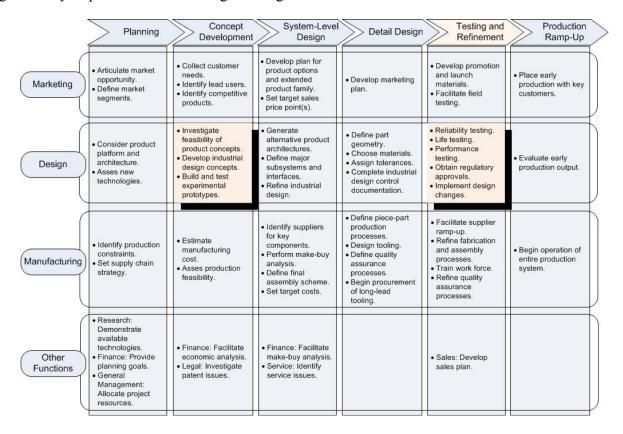


Figure 3-2: Phase-function view of the generic product development process

One should note that some experimental prototypes are expected to be built and tested prior to the detailed part geometry definition. These prototypes are thus based on lower maturity documentation created after the concept generation. They are named *looks-like* and *works-like* models and serve the purpose of concept illustration and testing: the formerly described alpha prototypes can sometimes fall in this category. Prototypes at the testing and refinement phase are some more mature approximations of the product since they disclose a great part of its actual behaviour and necessary changes assuming the level of approximation. Even though the main prototyping and testing activities are covered by the *Design* function as a whole, dedicated departments and defined interfaces still underlie the experimentation process in practice, as depicted in figure 3.3. In addition, specific transactions with manufacturing, procurement, production (shop-floor) and quality are deployed to effectively build the planned prototypes. Indeed, a critical aspect to consider regarding the testing and refinement phase is planning for prototypes (Ulrich & Eppinger, 2008). This planning stage can be divided into four steps as listed in figure 3.3. The whole experimentation process aims at optimising, on a lifecycle timeframe, the lessons learned that result from testing prototypes.

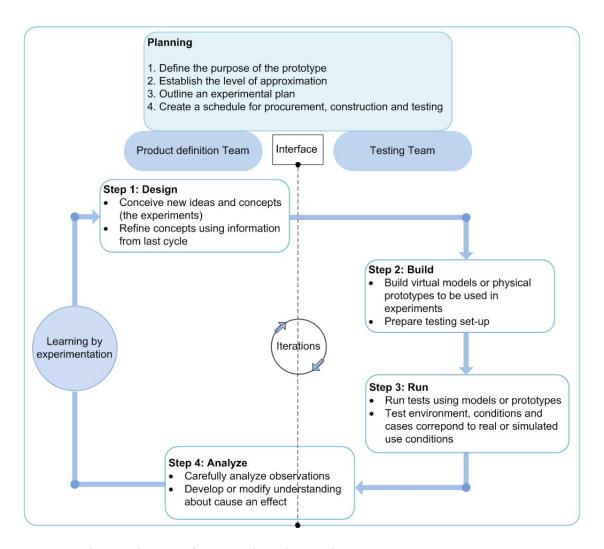


Figure 3-3: Experimentation as a four-step iterative cycle

As Thomke states that how firms link experimentation and testing activities to major process phases, system stages, and development tasks is an essential part of effective management practice (Thomke, 2008), the challenge related to the research presented here lies in managing interfaces in such a cross-functional context.

The next sections therefore discuss the main collaboration issues in a context where information and applications are scattered across functional teams.

## 3.3 Cross-functional collaboration in heterogeneous environments

Collaboration is a key challenge in industry especially during the product development stage where the following aspects are increasingly significant (Toche, et al., 2010):

- Early involvement of multiple partners (suppliers included);
- Incompatible systems and information structures;
- Communication and traceability;
- Product and process complexity and visibility.

A number of PLM solution providers have started to develop customised modules to overcome the native incompatibilities with the concurrent systems and, somehow, enable the collaborative work. However, these customisations are typically costly and to some extent still highly ineffective. This section describes information sharing and exchange in the light of conventions in general and with an insight into the role of languages and protocols for an effective communication between two or more entities. The approach is at the heart of the solution to address the situations involving internal and external partners working at diverse interfaces and using natively incompatible architectures to aggregate the common necessary information for an effective collaborative development.

#### 3.3.1 Models of communication

Two important aspects to consider in a collaborative product development environment are communication and coordination. In theory, five elements typically compose a communication process: a sender, a receiver, a channel, a code and the content. This telecommunication perspective of communications places more emphasis on the infrastructure for the communication than its intended impact, as suggested by Laswell's maxim *Who says what in which channel with what effect* (Laswell, 1948). However, even when two parties get involved in an exchange on a common channel and by selecting a specific code, it remains crucial to consider each of their *fields of experience* before concluding on the success of the content's transmission and the resulting usability (Schramm, 1954). To elaborate, the sender encodes the message based on its own *field of experience*. The receiver's *field of experience* directs decoding. If the sender and the receiver have nothing in common with regard to the *fields of experience*, then there is no communication happening (Croft, 2004). As a consequence, the extent to which the contents are decipherable and fully usable directly depend on the extent of the overlap of the two *fields of experience*. Figure 3.4 exhibits a model of communication to enhance PLM support in heterogeneous environments. It is inspired from the models of Schramm and Rachuri (Rachuri et

al., 2008; Schramm, 1954) and the influence of coordination mechanisms in this case of large distributed work (Thomson & Suss, 2009; Tichkiewitch & Brissaud, 2000) is inserted.

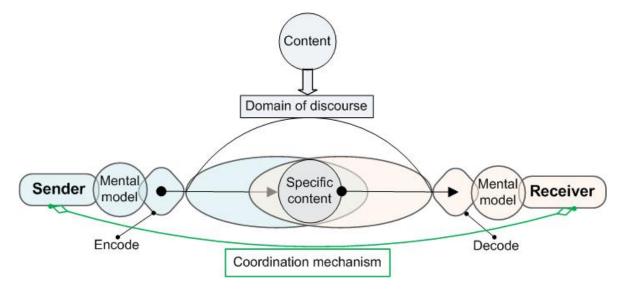


Figure 3-4: A model of communication in heterogeneous environments

The model can be transposed for the benefit of the PLM paradigm by considering the parties (sender and receiver) as the platforms of two participants, or partners when external. The mental models are equivalent to Application Programming Interfaces (API) and the ellipses overlapping in the centre represent the platforms' *fields of experience* that enable the sharing of specific contents within a certain domain of discourse (channel). Coordination mechanisms act here as triggers to pull or update information on each side according to the ongoing collaborative development activities. This model is suitable for PLM support at large because of the vast diversity of contents that could be exchanged or merged by adhering to common content specifications. The model is implemented in the next section through a demonstrative scenario of collaboration between two natively disparate PLM platforms.

## 3.3.2 A scenario of collaboration in heterogeneous environments

The following academic scenario is based on the use of Rialto Bridge, the outcome of an open source project included in the PLM lab open initiative (PLMlab, 2010). The scenario involves two partners (C and D) using different systems and collaborating on the development of a mold. The lead actor C has Windchill <sup>TM</sup> 9.0 as PLM system and Pro/Engineer Wildfire <sup>TM</sup> 3.0 as CAD system (PTC, 2010) and is to develop the upper side of the mold whereas partner D focuses on

the lower side and utilizes another renowned CAD software. The information packages produced by each partner should be continually aggregated in a central environment since many stakeholders use the assembled mold information in their daily tasks. Table 3.2 summarizes an infrastructure based on open initiatives and implanted to enable a collaboration scenario. Lead actor C's PLM system hosts the central environment and partner D is granted access to it.

Table 3.2: Key data for the collaboration scenario<sup>1</sup>

	Lead actor C	Partner D
PLM	Windchill <sup>TM</sup> 9.0 - Web-based solution enabling controlled external accesses.	Use logins and passwords to access the environment via any internet browser.
CAD	Pro/Engineer Wildfire TM 3.0	Other renowned CAD
Mapping to central environment	Intrinsic	Rialto Bridge (Open source)
Visualization and annotations	3DXML Player TM (free)	ProductView TM (Enabled in the central environment)

Within its PLM system, Partner C creates a project context corresponding to the isolated environment where the joint development will be carried out. The selected information is then made available (shared from internal contexts) in the environment while limiting exposure of source and other related data. It is done as such for confidentiality and security reasons. All team members are thereby invited and can use all enabled functionalities, particularly object creation and association, import/export and check-in/check-out within the environment. Partner D logs on to the environment and starts to share its work by adhering to the PLM schema, which means in this case mapping information as illustrated in figure 3.5 and taking part in the subsequent routines and processes.

<sup>1</sup> A detailed description of all the software tools listed in the table is available in (Toche, 2008).

\_

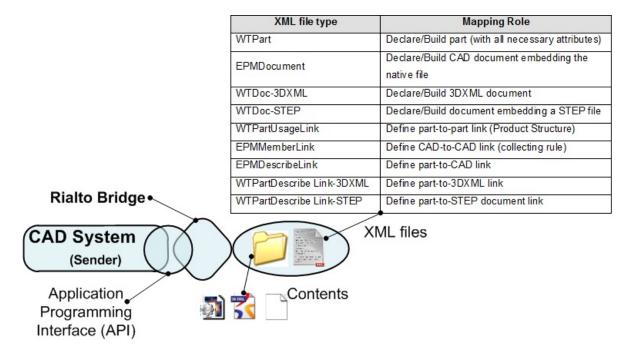


Figure 3-5: Mapping information to the central environment PLM schema with the transformation engine Rialto Bridge

The XML files and the specification data generated with the transformation engine follow an Import/Export schema to disseminate the product data within the central environment. It is through the automatic parsing of XML files in the environment that the structure is built. The 3D representations and documents are also attached, the attributes are set within the environment and the DMU (as defined in Section 2.3) is therefore shared.

Once imported into the project, the two structures are associated and the spatial information is reconciled manually in a CAD assembly by using the STEP-AP214 (ISO10303-11, 2004) main file from partner D. This provides a unique view of the assembled mold for the two partners; partner D using the enabled 3D player within the central environment. No mapping operation was found to actually make the central environment automatically recognize incoming STEP (ISO10303-11, 2004) documents as manageable CAD documents and accordingly use the specific functionalities related to those items. The aforementioned manual operation to reconcile the product representations was therefore indispensable. Figure 3.6 shows the cross visualizations and the unified structure within the central environment.

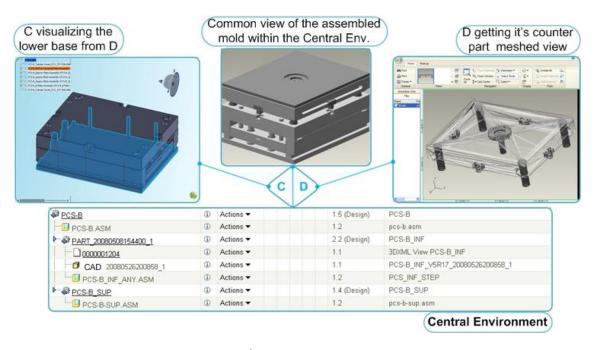


Figure 3-6: Unified structure of the product<sup>2</sup> and common and cross visualizations

Although the initial import and association of the product structures from C and D seem to offer an acceptable way of merging the two work packages in a unique DMU with consistent metadata, the process turns awkward when modifications start. This is mostly caused by the unavailability of CAD integration mechanisms for the imported CAD documents in the central environment. In fact, during a modification on partner D's side, checked-out (or reserved) documents are locally downloaded on D's platform instead of being managed within the vault, thus preventing the environment from automatically applying iterations.

Each change on D's side therefore necessitates a new version-controlled mapping and import in the central environment to make up-to-date information available to the whole team, which is cumbersome and can lead to inconsistencies. The Rialto Bridge open source project is currently addressing this concern (PLMlab, 2010).

Nevertheless, critical issues in interfacing silo-arranged departments in the extended enterprise are addressed in the scenario. As the scenario indicates, it is crucial to delimit the domain of discourse and carefully establish a content specification before collaborators can succeed in this sort of exchange. Hence, the next chapter discusses an appropriate framework that

<sup>&</sup>lt;sup>2</sup> The 3D models are available in diverse formats on 3DContentCentral for free (3DContentCentral, 2010).

leverages configured engineering product information, based on complementary information structures, to support the management of prototyping and testing information in a PLM perspective. It also introduces a neutral exchange environment to cope with the observed challenges when it comes to natively incompatible tools, silo-arranged departments and client-supplier relationships.

## CHAPTER 4 A TWO STEP FRAMEWORK FOR SHARING PROTOTYPING AND TESTING INFORMATION

As observed in chapter one, the sharing and exchange of prototyping and testing information within the configured engineering product information context will mainly be achieved by resolving the two-fold problem reiterated below:

- **P1**: the required physical tests and procedures, and the derived *as-built* structures are not systematically connected to the *as-designed* structure;
- **P2**: hardware testing transactions and prototype information tracking are not addressed within common PLM visions.

The first section of this chapter presents a practical example centred on an aircraft pylon, with emphasis placed on the forward engine mount, as a case study meant to carry out the design/simulation of the targeted framework. Figure 4.1 displays the exploration pattern.

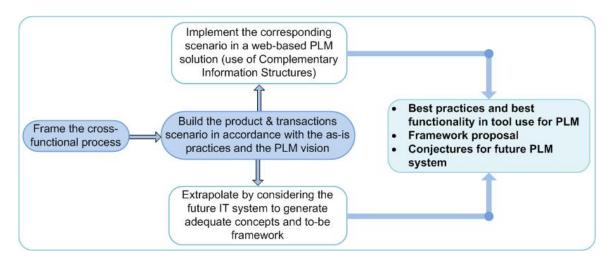


Figure 4-1: Steps for the design of the PLM framework supporting the identified transactions

The activities in the two first rounded boxes are completed by relying on the case study, as the real life processes and transactions are also replicated. The P1 and P2 concerns are then explored separately with the interoperability issues considered for the second (lower box). The inferences and outcomes are synthesised in the last box.

The use of complementary information structures and configuration methodologies, to track physically built instances<sup>3</sup>, is discussed in the second section of the chapter. Implementing such a combined approach is seen as the most appropriate way to cope with the P1 concern. This activity is presented in figure 4.1 in the upper box. The outcome from carrying out the scenario in the project's PLM system is hence discussed to:

- Capture the best practices and best functionality in tool use for PLM.
- Consolidate the framework proposal.
- Conjecture the future PLM systems.

The Open Exchange Nest (OEN) is finally introduced to address the P2 issue and, to a large extent, incorporate up and downstream lifecycle transactions. This is done as part of the activity presented in the lower box in figure 4.1.

#### 4.1 The retrofit of an aircraft engine: A case study

A simplified example of information flow in a development and testing department is presented here to highlight its cross-functional interfaces and to better understand the transactions in terms of data and material inputs and outputs. This example is drawn both from interviews conducted with the industrial partners within the PLM2 project (PLM2, 2010) and the virtual environment student project held each year at École Polytechnique de Montréal (Fortin et al., 2006). The case study features a development team involved in a significant engineering change consisting of the design of a new pylon to install the PW305A engine from Pratt & Whitney Canada on the Bombardier Aerospace CRJ-700 regional jet. The retrofit provides a new variant of the aircraft and the change necessitates re-establishing compliance with aviation regulations. This basically means carrying out certification tests and presenting analysis reports concerning the main subassemblies, being the pylon's main structure and secondary structure, along with the bleed air, FIREX, fuel, hydraulic and electrical systems. The focus here is on the forward engine mount, which forms part of the pylon's main structure and for which the FAR25 certification chapter

<sup>&</sup>lt;sup>3</sup> Not to be confused with an occurrence, an instance corresponds to the materialisation of an artefact and therefore identifies a unique physical part, whereas the occurrence identifies the presence and unique position of a part within the digital mock-up.

applies. As for any regulated aircraft design process, the planning for prototyping work is done at the Advanced Concept Review (ACR) and during the elaboration of the General Compliance Plan and the Certification plan.

The forward engine mount is the design artefact to be prototyped, more or less approximately, in the role, implementation and/or look and feel dimensions. The role of this mount is to support the engine in take-off, flight and extreme crash conditions; to transmit the engine thrust to the aircraft; and to form a barrier which defines the fire zone. The implementation is therefore restrained to the selection of an improved titanium alloy and the structural design itself. The look and feel is of no interest in this case. The *as-designed* mount assembly in the context of the pylon's configured DMU is displayed in figure 4.2.

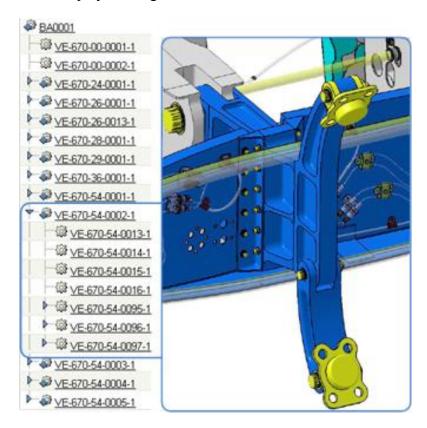


Figure 4-2: As-designed forward engine mount and corresponding 3D representation in the context of the pylon's configured DMU

The mount is a core artefact within the pylon structural design and is to withstand severe loads, vibrations and possibly fire. Along with this, the mount is a true interface component since it is the link between the installed engine and the fuselage frame. In addition, the connecting

interface with the engine has to be as flexible as possible for the engine installation and to sustain its slight deviations during some flight stages (see thermal expansion and other phenomena). The mount therefore features both an upper and lower pad and a mount link as enablers of the free moving fixations. The hardware labelled C in figure 4.3 includes three bearings to allow the pivoting movements, and some titanium fasteners to strengthen the connections.

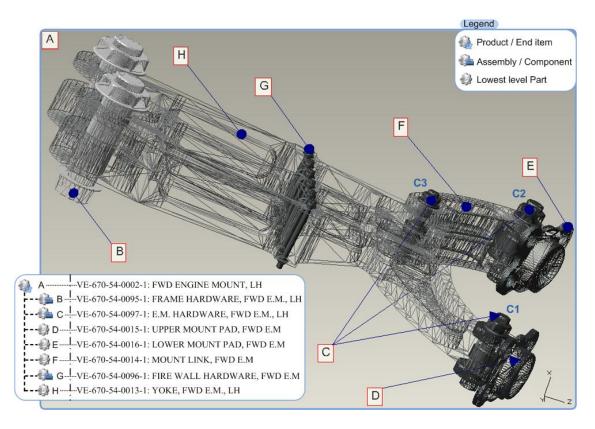


Figure 4-3: A Design arrangement of the forward engine mount components

In conformity with the functional view on this specific end-item, designers have provided the displayed *as-designed* BOM. It must be noted how assemblies and components are defined strictly by following the functional logics, as it is the case for the hardware components B, C and G. One should also notice that the lower level parts composing the three listed assemblies are not displayed within the BOM in the figure in order to ease readability.

As validated by an Integrated Product Team (IPT), weight reduction efforts are noticeable on this complex, five axis machined part. Manufacturing time and cost of this primary component are also optimized. The team in charge of the structural tests defines the prototyping/manufacturing strategy. Furthermore, the team has decided to outsource the

machining of the parts and to use an external partner's test facilities. According to the level of approximation, the prototyped parts are sometimes not identical to *as-designed* models since modifications could be done for instrumentation reasons or simply because the test on a specific physical instance is to explore a particular behaviour of the component.

According to the prototyping/manufacturing strategy, the testing team should therefore manage, in parallel and when necessary, local BOMs mirroring and following *as-designed* versions. The general information on a manufactured and assembled instance to be shipped for tests is organized in a test plan previously validated both by the design and testing teams. The test plan, the test procedure for its execution, and the results represent the main outcomes of the testing activities that will be attached to the complementary structure and tracked to refine the artefact.

Regarding the modifications that lead to *as-built* arrangements adequate for tests, one should note the similarities with the transactions to build and maintain the mBOM usually deployed through the MPM module. Indeed, the manufacturing product structures also result from the manufacturing strategies and are rarely identical to *as-designed* structures. In addition and as further illustrated, the structures remain interconnected through the MPM module which basically uses the notions of (Fortin et al., 2010; Huet, et al., 2009):

- **Equivalence link** to relate a part iteration in the mBOM to the equivalent part in the eBOM and thus ensures conformity and traceability.
- **Occurrence link** to relate the position of equivalent parts within the engineering and manufacturing BOMs and thus enable the view of the identical mock-up.
- **Reference link** to propagate change when a part iteration on the manufacturing side doesn't have a strict equivalent on the engineering one.

By relying on these links, the pattern through which a manufacturing structure is built and maintained in as-is platforms and from a lifecycle management perspective can be described. This description is elaborated in the next section in order to demonstrate how the P1 concern is equally solvable by these means.

## 4.2 Step one: Building and maintaining complementary and configurable structures

According to the project milestones and the design maturity, the two processes that are Design and Process Plan concretely begin to overlap at a certain point and a great part of the Manufacturing activities then start after the full access to the product definition data. These data are clustered within the configured DMU (as previously defined). They can then be filtered and manipulated by following the effectivity rules within the agreed upon configuration management context. Hence, while obtaining the same functional arrangements as the designers, the manufacturing (idem development & testing) engineers have to use their expertise to actually deploy strategies to build, assemble and test the artefact based on the shop floor resources and constraints.

Regarding the forward engine mount, its manufacturing has been found to necessitate a quite different breakdown structure from the *as-designed* one. This manufacturing structure indeed follows a strict chronology of operations to end-up with the physically functioning artefact. To elaborate, two sets of operations (the pressing of the bearings and the fastening being the principal in both) have to be followed to obtain:

- On one hand, the manufacturing assembly MFG-ASSY-001 which includes the upper mount pad, the yoke and the hardware C1.
- On the other hand, the manufacturing assembly MFG-ASSY-002 including the lower mount pad, the mount link and the hardware C2.

These two are then brought together within the manufacturing assembly MFG-ASSY-003 by using the hardware C3; the bearing being pressed into the yoke as a prior operation. In a final operation, all the other components are installed and the physical assembly of the mount is therefore considered achieved. Figure 4.4 exhibits the resulting breakdown structure. The complementary structure and its links to the *as-designed* one are maintained throughout the Design and Manufacturing (idem Development & Testing) iterations. In fact, the described links are not static but rather dynamic and are effectively updated, when necessary, to keep stakeholders on the track.

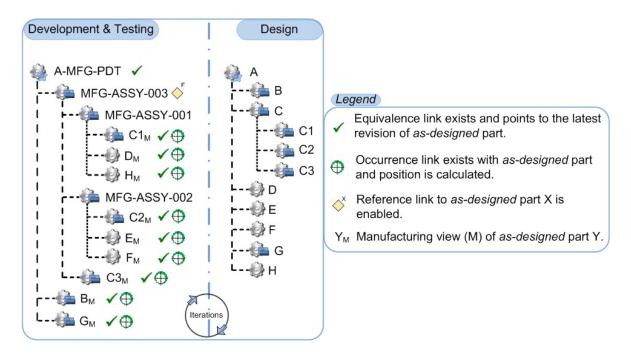


Figure 4-4: Manufacturing complementary structure of the forward engine mount

The mount link F is a connecting part that is susceptible to undergo several minor and major changes on the yoke (H) side during the development. Any change on the link may directly affect the manufacturing assembly MFG-ASSY-003 and then trigger minor or major changes to the related process plan; this includes changes to the standard bearing and fasteners composing the hardware C3. Activating the reference link to the mount link F on the manufacturing assembly MFG-ASSY-003 is therefore necessary to track and propagate the eventual changes.

Generally, as depicted in figure 4.5, the user interface that serves to build and maintain the complementary structure is quite similar to figure 4.4, and it features complementary panels allowing easy drag and drop, fast search, view and link of data. Also, a full CAD visualisation tool is provided to obtain 3D representations and mock-ups either generated from the engineering BOM or the manufacturing one. The visualisation tool is further used to identify the right occurrences of the parts, when multiple, and reciprocally select them within the BOMs. Last and not least, the tool serves to make all the annotations during process sheet design. Hence, the process sheets populate the mBOM as part of the process plan along with the resources and time allocation tables, as visible when the upper right tabs in the figure are enabled – see figure 4.5 on the next page.

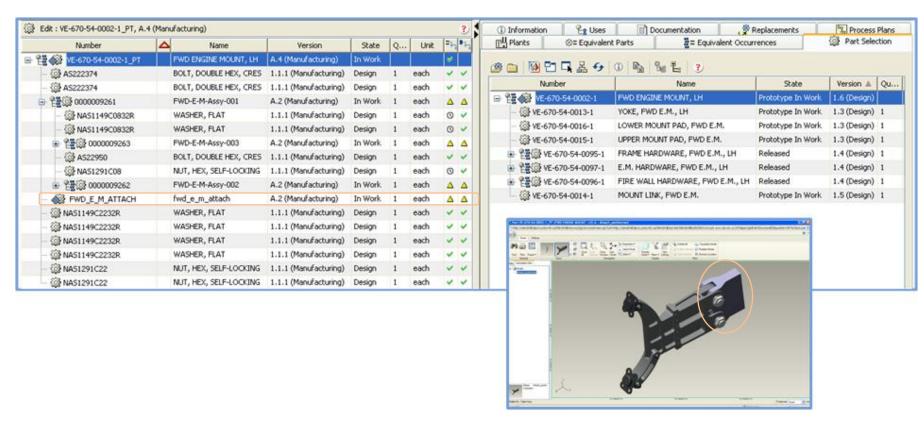


Figure 4-5: Building and maintaining complementary structures: A user interface - Courtesy of PTC (PTC, 2010)

As seen in figure 4.5, a whole new component, the y-shaped attachment has been added by Development & Testing as part of the test rig to carry out the structural test. Indeed, when it comes to some development activities like the validation of manufacturing processes or those of prototyping and testing, several functionalities are available in addition to what has already been mentioned. These include, among others, the adding of components specifically designed for the tests, e.g.: instrumentation, test rigs, etc... or simply the fast reuse of some portion of structures throughout several test cases.

Assuming that *as-built* structures are deployed and remain connected to the *as-designed* one as previously discussed, figure 4.6 illustrates how each physically built instance is then mirrored and tracked both by Design and Manufacturing (idem Development & testing). To elaborate, once a product structure is constructed, either being the eBOM (functional arrangement) or the mBOM (Manufacturing/Testing strategy), the same configuration methodology is followed to generate frozen BOMs that rely on lot and manufacturing serial numbers to mirror each physically built and tested instance.

Figure 4.6, as displayed on the next page, features a simplified example that is used to ease readability. The figure is centred upon a conceptual end-item labelled A which includes two parts, C and D, and an assembly B composed of two parts B1 and B2. The described pattern is the same that has been followed for the forward engine mount case study – see appendixes A and B for more details on the simulation in the PLM system Windchill 9.1<sup>TM</sup>.

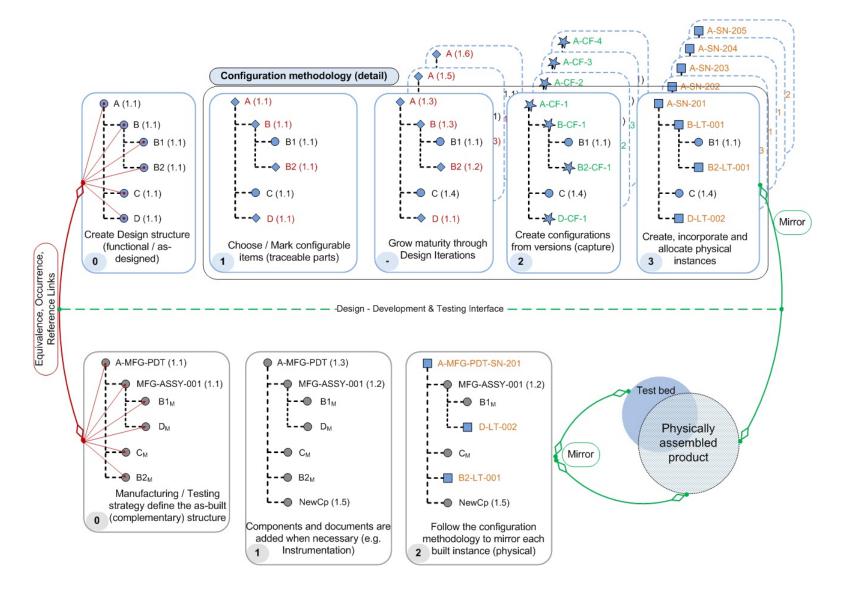


Figure 4-6: Complementary and configurable information structures - a diagrammatic analysis

So once a structure is created, or during its creation, the methodology consists of:

- Selecting and marking the relevant parts that have to be tracked throughout the lifecycle. These parts are designated as traceable parts and the trace code can be a serial number, a lot number or a combination of both.
- Creating configurations for traceable parts whenever a significant maturity is reached this works as capturing or taking a snapshot of the corresponding versions.
- Creating instances from configurations: an instance, which requires a serial number, is to correspond to an existing physical part, such as one manufactured for tests, at the appropriate level of maturity.
- Incorporating lower level instances: incorporation is the date when a new configuration associated to an end item instance takes effect (PTC, 2008). The method is used to track ongoing configuration changes to an end item instance that has previously been tested or is in service.
- Allocating instances: allocation is the process of associating specific end item instances and serialized parts to each other (PTC, 2008). In fact, a top-level end item instance is not completely defined until all of the serialized parts and end item instances are associated with it.

A prototype could therefore be, for example, built and tested from a specific configuration of the end-item while Design iterations are continuing and evolving. To do so, instances of the parts are generated from the captured configurations and identified by their serial numbers or lot numbers as provided by the suppliers or the shop-floor. These instances are allocated to form the complementary structure (*as-built*) mirroring the physically assembled prototype. Since the structure remains linked to its *as-designed* version, the traceability regarding all the approximations made while prototyping and testing is enabled. Hence, because the manufacturing serial and lot numbers are identified, the physical parts are also tracked. Of course this methodology could also be applied when all parts and documents have reached the release status. This is to say that each tested, produced or delivered instance of the end-item is fully traceable by the preceding means.

Even though the testing transactions and the prototype information tracking is not explicitly addressed within as-is PLM platforms, this new approach has been implemented within the project's PLM system as a simulation to converge towards the targeted framework. However, it has verified that the required physical tests and procedures (transactions and documentation) and the derived *as-built* structures can thoroughly be connected to the *as-designed* structure. In addition, the P2 concern is therefore implicitly and partially addressed within a tangible PLM context; the remaining aspect to consider being how to cope with software interoperability issues in silo-arranged engineering and testing teams in general. Addressing this issue would allow one to end up with a framework proposal that both meets the P1 and P2 concerns while fully implementing the integrated cooperative and collaborative management of *product data*, throughout the entire product life as prescribed in the PLM vision.

# 4.3 Step two: Facing software incompatibilities – the open exchange platforms

The team in charge of the structural tests has decided to outsource the machining of the parts and to use an external partner's test facilities. The case study then features the already mentioned challenge of the interoperability of tools during cross-functional collaboration between siloarranged departments or within client-supplier relationships.

The scenarios and observations made in Chapter 3 have tangibly highlighted the hurdles in the course of collaborating when using natively disparate architectures or simply evolving in a heterogeneous environment. In search of the interconnection and as part of the extrapolation mentioned in the steps for the design of the targeted PLM framework (see figure 4.1), open exchange environments are considered here for two main reasons:

- Multi-partner development projects will increasingly take place;
- The real cost of the lack of interoperability is not measurable.

Descriptions of open design web portals (Koch & Tumer, 2009; Tucker & Shumacher, 2009) are appearing in the literature, and the main areas to be addressed and integrated in a web framework are documented. Some are cited below:

- Project dashboard: user customizable page grouping relevant design project information and threads;
- Documentation and Design repository: detailed project information storage;
- Communication: infrastructure for individual and team communication;
- User identification standard: secure and standard log in for members;
- Licensing: designer's IP protection.

However, regarding how different the content reference models are from one department to another or from one PLM solution provider to another, a common denominator should be sought to make them dialogue in a straightforward manner. This will only be enabled by adhering to a model of communication as illustrated in Figures 3.4 and 3.5, where mapping information in federated content specifications is a key to the success of the exchange. In addition, it is through federated standards, languages and protocols that collaboration will effectively happen during the distributed product development in a PLM perspective (Rachuri, et al., 2008; Rosén, 2006; Song & Chung, 2009).

The concept of the Open Exchange Nest is introduced here to describe an environment where multiple partners, regardless of their tools, could merge product and process data and interoperate seamlessly to effectively develop a product in a network configuration and a lifecycle management framework. Figure 4.7 illustrates the envisioned shift. The XML Schema is used for the content specification and the Simple Object Access Protocol (SOAP) serves to securely flow contents. In combination with the latter, the Web Services Description Language (WSDL) is used to list the collections of network endpoints and to perform the supported transactions. By relying on web services, the neutral sharing, exchange and management of the data is enabled over the Internet.

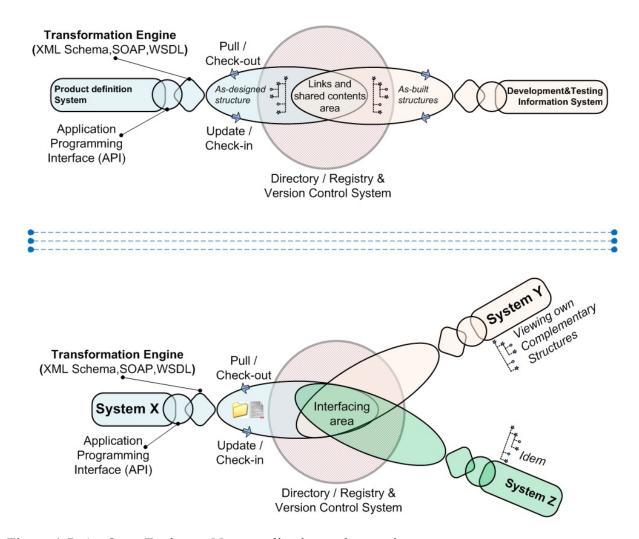


Figure 4-7: An Open Exchange Nest: application and extension

As discussed in section 3.3.1, each *System* in the lower part of figure 4.7 should be regarded as any participant's PLM tool or any stakeholder's dedicated tool (maintenance, quality, procurement, etc.). The transformation engine provides the encoding-decoding methods and therefore the means for the exchange of the contents that are to be aggregated in (and retrieved from) a central environment. The overlapping in the centre does not represent a master model but rather an area where the systems can interface to link the complementary information structures and enable the collaborative work. As interfacing methods are used herein, the concept is rather towards *knowledge integration* than straight *knowledge fusion*. The first being a collection of independent knowledge systems with clearly defined interfaces, whereas the second aims to create a new knowledge system that can be operated as a whole to develop multi-disciplinary products (Tomiyama & Meijer, 2006).

#### CHAPTER 5 CONCLUSIONS AND FUTURE WORK

The role of DMUs and the specificity of information in PLM have primarily been presented in Chapter 2 to illustrate the context in which products are now developed in the modern enterprise. As such, an insight into the usage of virtual prototypes and DMUs has been provided in sections 2.1 and 2.3 and the correlation between simulation and physical tests has thoroughly been discussed in section 2.2. This has been done to emphasise the growing trend to organise testing information so as to be linked to *as-designed* structures and to mirror the built prototypes within the pre-existing PLM infrastructures.

The role and place of the testing and refinement phase during the PDP have also been discussed in Chapter 3. The key aspects concerning prototyping and testing activities have been presented in sections 3.1 and 3.2 to demonstrate the interoperability needed between design and testing departments or, in a broader context, between teams involved in the process of learning by experimentation. Issues with respect to collaboration by means of heterogeneous infrastructures have also been raised in section 3.3 and a model of communication has been implemented in a collaborative scenario. The demonstration has shown how natively disparate tools can dialogue and merge information in a central environment while adhering to modern PLM requirements. This demonstration forms a basis from which one can address the software incompatibilities as tackled in the second step of the two-step framework. Furthermore, the shift to the intended PLM framework has been drawn from interviews with the industrial partners of the PLM2 project and the virtual environment student project held at École Polytechnique de Montréal.

A case study based on the retrofit of an aircraft engine has then been deployed in sections 4.1 and 4.2 to implement a scenario disclosing the functionalities to be available within a homogeneous framework. These functionalities ensure that:

- The required physical tests and procedures and the derived *as-built* structures are systematically connected to the *as-designed* structure.
- The hardware testing transactions and prototype information tracking are addressed in conformance with the PLM vision within industry and as described in the literature.
  - Windchill 9.1<sup>TM</sup>, a state of the art PLM system, has been used for this implementation.

In accordance with the domains of expertise involved in prototyping and testing activities and the current practices in industry, it has been demonstrated that the complementary information structures approach represents the most suitable way of coping with the listed concerns. The limitation of this work remains in demonstrating that the transactions can be carried out in a real life situation with teams made up of a large number of participants. Also, no systematic method to transpose this implementation in other existing homogeneous PLM systems was proposed. However, all the principles underlying the implementation were thoroughly explained on a system neutral basis to ease any eventual transposition.

Regarding the increasing involvement of multiple partners in all phases of the PDP and the native incompatibilities between the specialized tools used in different domains of expertise, the Open Exchange Nest concept has been introduced in section 4.3 as a recommendation. The concept provides a framework where complementary information could be mapped and exchanged neutrally by adhering to common content specifications. To elaborate, the challenge in enabling such a framework that is ideal for the identified cross-functional transactions is two-fold. On one hand there is this need to systematically connect the diverse stakeholders' structures to yield the efficient collaborative environment that matches lifecycle management requirements (the focus of step one). On the other hand, there are software interoperability issues that don't only affect Design – Testing interactions but many other interfaces within either the company – due to the silo-arrangement – or consortiums of partners, in which case the whole PLM platforms could simply be incompatible (the focus of step two). The Open Exchange Nest (OEN) is therefore presented as a generic PLM-driven and web-based concept to support the collaborative work in the aforementioned context.

Three directions for future work are discussed below and they focus on validating the results from the current research work on a realistic scale as well as extending the application of the drawn methods.

- Implement step one approach (complementary and configurable information structures) in a real life situation. This could be done for example by selecting one less critical product of the portfolio and having the product definition team and the test team following the described methodology until the production ramp-up. However, the two teams should both work in a homogeneous PLM system as it is a core assumption in step one approach. Such an

- implementation may consolidate the approach and raise the diverse issues when a large number of participants are involved.
- Provide a systematic model to better encapsulate step one approach and therefore ease eventual transpositions enfolding some other PLM systems. This work should take advantage of the system neutral analyses done herein.
- Simulate a scenario adhering to the Open Exchange Nest approach. As the OEN deals with interoperability issues in general, this simulation can also be done wherever an exchange of information is needed between stakeholders working on BOMs basis. The simulation will first emphasise on specifying the federated content (what to exchange, in what format?), and defining the XML schema for the mapping. The work will then consist of using each stakeholder's system API to build each transformation engine. The directory/registry and version control system should be deployed in parallel as it is the central environment where the data processed by the transformation engines are stored and managed. Once the exchange is enabled, the XML schema will be gradually enhanced so as to completely enable the viewing and linking of both generic data and complementary structures.

### LIST OF REFERENCES

- 3DContentCentral. (2010). Free 3D Models, Free CAD models. Retrieved April, 3rd 2010, from <a href="http://www.3dcontentcentral.com/">http://www.3dcontentcentral.com/</a>
- ANSI/EIA-649-A. (2004). National Consensus Standard for Configuration Management:

  American National Standards Institute.
- Berchtold, G. (2000). Digitales Mockup und virtuelles Flugzeug. Presentation of DaimlerChrysler Aerospace Military Aircraft: Technical University of Munich.
- Brissaud, D., & Tichkiewitch, S. (2000). Innovation and manufacturability analysis in an integrated design context. *Computers in Industry*, *43*(2), 111-121.
- Brissaud, D., & Tichkiewitch, S. (2001). Product models for life-cycle. *CIRP Annals Manufacturing Technology*, 50(1), 105-108.
- Carleton, T., & Cockayne, W. (2009). *The power of prototypes in foresight engineering*. Paper presented at the Proceedings of the International Conference on Engineering Design, ICED'09, Stanford, CA, USA.
- Clark, K. B., & Fujimoto, T. (1991). Product development performance: strategy, organization, and management in the world auto industry Boston, Massachusetts: Harvard Business School Press.
- Coyle, J. M., & Paul, D. (1997). Virtual Product Definition The Key to Affordable and Supportable Products. Paper presented at the AGARD Symposium on "Future Aerospace Technology in the Service of the Alliance", Palaiseau, France.
- Croft, R. S. (2004). Communication Theory. La Grande, OR: Eastern Oregon University.

- Cuillere, J.-C., Deneux, D., & Maranzana, R. (1997). Integrated CAD and CAPP around feature recognition techniques. In P. Chedmail, J.-C. Bocquet & D. Dornfeld (Eds.), *Integrated design and manufacturing in mechanical engineering*. Dordrecht: Kluwer Academic Publishers.
- Danesi, F., Gardan, N., Gardan, Y., & Reimeringer, M. (2008). P4LM: A methodology for product lifecycle management. *Computers in Industry*, 59(2-3), 304-317.
- Dolezal, W. R. (2008). Success Factors for Digital Mock-ups (DMU) in complex Aerospace

  Product Development. Technische Universität München, Genehmigten Dissertation,

  München.
- Döllner, G., Kellner, P., & Tegel, O. (2000). Digital Mock-up and Rapid Prototyping in Automotive Product Development. *Journal of Integrated Design and Process Science*(March 2000), 55-66.
- Drieux, G., Leon, J. C., Guillaume, F., & Chevassus, N. (2006). *Integrating design with downstream applications through product shapes adaptation processes*. Paper presented at the IMACS Multiconference on "Computational Engineering in Systems Applications" (CESA), Beijing, China.
- Fortin, C., & Huet, G. (2007). Manufacturing Process Management: iterative synchronisation of engineering data with manufacturing realities. *Int. J. Product Development, Vol. 4*(n° 3/4), pp. 280-295.
- Fortin, C., Sanshagrin, B., Huet, G., & Gagné, S. (2006). *The CAMAQ project: a design-build experience based on a virtual immersion in aerospace industry practices*. Paper presented at the Proceedings of the 2nd CDIO conference, Linköping, Sweden.

- Fortin, C., Toche, B., McSorley, G., & Huet, G. (2010). Supporting Product Development Value Streams through Manufacturing Process Management. Paper presented at the Proceedings of the 1st International Conference on Modelling and Management of Engineering Processes, MMEP'10, Cambridge, UK.
- Gausemeier, J., Lindemann, U., Reinhart, G., & Wiendahl, H. (2000). Kooperatives

  Produktengineering Ein neues Selbstverständnis des ingenieurmäßigen Wirkens (Vol.
  79). Paderborn: Heinz Nixdorf Institut Verlagsschriftenreihe, Universität Paderborn
  2000.
- Gerber, E. (2009). *Prototyping: Facing uncertainty through small wins*. Paper presented at the Proceedings of the International Conference on Engineering Design, ICED'09, Stanford, CA, USA.
- Grieves, M. W. (2005). Product lifecycle management: the new paradigm for enterprises. *Int. J. Product Development, Vol.* 2(1/2), pp. 71-84.
- Houde, S., & Hill, C. (1997). What do prototypes prototype? In M. Helander, T. Landauer & P. Prabhu (Eds.), *Handbook of Human-Computer Interaction* (2nd ed.). Amsterdam: Elsevier Science B. V.
- Huet, G., Fortin, C., McSorley, G., & Toche, B. (2009, May 13-15, 2009). *Information structures* and processes to support data exchange between product development and resource planning systems. Paper presented at the Proceedings of the IESM'09 International Conference on Industrial Engineering and Systems Management, Montréal, Canada.

- ISO10303-11. (2004). Industrial automation systems and integration Product data representation and exchange Part 11: Description methods: The EXPRESS language reference manual: International Organization for Standardization.
- Jarratt, T., Clarkson, J., & Eckert, C. (2005). Engineering change. In J. Clarkson & C. Eckert (Eds.), *Design process improvement : a review of current practice* (pp. 262-285). London, UK: Springer.
- Kim, B.-Y., & Bum-Kyu, K. (2008). Cross-functional cooperation with design teams in New Product Development. *International Journal of Design*, *2*(3), 43-45.
- Koch, M., & Tumer, I. Y. (2009). *Towards Open Design: The emergent face of engineering*.

  Paper presented at the Proceedings of the International Conference on Engineering Design, ICED'09, Stanford, CA, USA.
- Laswell, H. (1948). The structure and function of communication in society. In L. Bryson (Ed.), *The communication of ideas*. New-York: Harper.
- Lazzari, D., & Raimondo, G. (2001). *Digital Mock-Up in support to Space Station elements Integration and Test.* Paper presented at the Proceedings of the 4th International Symposium on Environmental Testing for Space Programmes, Liege, Belgium.
- Liu, W., Zeng, Y., Maletz, M., & Brisson, D. (2009). Product lifecycle management: A survey.
   Paper presented at the Proceedings of the ASME 2009 International Design Engineering
   Technical Conferences and Computers and Information in Engineering Conference,
   IDETC/CIE 2009, San Diego, California, United states.
- Maropoulos, P. G. (2003). Digital enterprise technology defining perspectives and research priorities. *International Journal of Computer Integrated Manufacturing*, *16*(7), 467-478.

- McMahon, C., & Browne, J. (1993). *CAD/CAM From principles to practice*. Wokingham: Addison-Wesley Publishing Company.
- McSorley, G., Huet, G., & Fortin, C. (2008). A preliminary framework for the Management of "Product In-Use" information. *Collaborative Design and Numerical Engineering*
- Negroponte, N. (1995). The DNA of Information *Being digital*. New york: Knopf.
- Panchenko, Y., Moustapha, H., Mah, S., Patel, K., Dowhan, M. J., & Hall, D. (2002). *Preliminary multi-disciplinary optimization in turbomachinery design*. Paper presented at the RTO AVT Symposium on "Reduction of Military Vehicle Aquisition Time and Cost through Advanced Modelling and Virtual Simulation", Paris, France.
- PLM2. (2010). Collaborative development for Product Lifecycle Management. Retrieved April, 3rd 2010, from <a href="http://www.mcgill.ca/plm2-criag/">http://www.mcgill.ca/plm2-criag/</a>
- PLMlab. (2010). Applied Research in Product Lifecycle Management. Retrieved April, 3rd 2010, from <a href="http://www.plmlab.fr/">http://www.plmlab.fr/</a>
- PTC. (2008). Introduction to Windchill PDMLink 9.0/9.1 (Vol. T2136-090-02). Needham, MA, USA: Parametric Technology Corporation Global Services.
- PTC. (2010). Parametric Technology Corporation. Retrieved April, 3rd 2010, from <a href="http://www.ptc.com/">http://www.ptc.com/</a>
- Rachuri, S., Subrahmanian, E., Bouras, A., Fenves, S. J., Foufou, S., & Sriram, R. D. (2008). Information sharing and exchange in the context of product lifecycle management: Role of standards. *CAD Computer Aided Design*, 40(7), 789-800.

- Rangan, R. M., Rohde, S. M., Peak, R., Chadha, B., & Bliznakov, P. (2005). Streamlining Product Lifecycle Processes: A survey of product lifecycle management implementations, directions, and challenges. *Journal of computing and information Science in Engineering*, 5(3), 227-237.
- Riel, A., & Brenner, E. (2004). *Simulation Interoperability in Product Development*. Paper presented at the Proceedings of the 2004 European Simulation Interoperability Workshop (EURO-SIW), Edinburgh, Scotland.
- Rosén, J. (2006). Federated through-life support, enabling online integration of systems within the PLM domain. Paper presented at the Proceedings of the 1st Nordic Conference on Product Lifecycle Management NordPLM'06, Göteborg, Sweden.
- Ryan, R. R. (1999). Digital Testing in the Context of Digital Engineering Functional Virtual Prototyping, *VDI Berichte* (Vol. 1489). Berlin, Germany.
- Sääksvuori, A., & Immonen, A. (2004). *Product Lifecycle Management* Berlin: Springer-Verlag.
- Schramm, W. (1954). How communication works. In W. Schramm (Ed.), *The process and effects of mass communication*. Urbana, IL: University of Illinois Press.
- Shah, J. J., & Mäntylä, M. (1995). *Parametric and feature based CAD/CAM*. New york, NY: Wiley-Interscience Publication, John Wiley Sons Inc.
- Song, I.-H., & Chung, S.-C. (2009). Synthesis of the digital mock-up system for heterogeneous CAD assembly. *Computers in Industry*, 60(5), 285-295.
- Stark, J. (2005). Product Lifecycle Management, 21st Century paradigm for product realisation (2 ed.). London: Springer

- Szykman, S., Fenves, S. J., Keirouz, W., & Shooter, S. B. (2001). A foundation for interoperability in next-generation product development systems. *CAD Computer Aided Design*, 33(7), 545-559.
- Thomke, S. (2008). Learning by experimentation: Prototyping and testing. In C. H. Loch & S. Kavadias (Eds.), *Handbook of new product development management* (pp. 401-420). Amsterdam: Elsevier: Butterworth-Heinemann.
- Thomson, V. J., & Suss, S. (2009). *Coordination in New Product Development: A Model to Evaluate Potential Benefits*. Paper presented at the Proceedings of the 6th International Conference on Product Lifecycle Management, University of Bath, Bath, UK.
- Tichkiewitch, S., & Brissaud, D. (2000). Co-ordination between product and process definitions in a concurrent engineering environment. *CIRP Annals Manufacturing Technology*, 49(1), 75-78.
- Toche, B. (2008). Bridging CATIA V5 and Windchill 9.0. / Évaluation et développement du pont de transfert CATIA Windchill. École Polytechnique de Montréal, Montreal, QC, Canada.
- Toche, B., Huet, G., McSorley, G., & Fortin, C. (2010). *A Product Lifecycle Management Framework to Support the Exchange of Prototyping and Testing Information*. Paper presented at the Proceedings of the ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, IDETC/CIE 2010, Montreal, Canada.
- Tomiyama, T., & Meijer, B. (2006). Directions of Next Generation Product Development *Advances in Design* (pp. 27-35).

- Tucker, J. M., & Shumacher, M. (2009). *Moving new venture new product development from information push to pull using web 2.0.* Paper presented at the Proceedings of the International Conference on Engineering Design, ICED'09, Stanford, CA, USA.
- Ulrich, K. T., & Eppinger, S. D. (2008). *Product design and development* (4th ed.). Boston, Massachusetts: McGraw-Hill Higher Education.
- Van Der Auweraer, H., & Leuridan, J. (2005). A new testing paradigm for today's product development process Part 1. *Sound and Vibration*, *39*(9), 14-18.
- Warsh, D. (1998). What drives the wealth of nations? *Harvard Business Review*, 76(4), 171-175.
- Waurzyniak, P. (2008). PLM tools speed developments. *Manufacturing Engineering*, 140(6), 101-108.
- Waurzyniak, P. (2010). Speeding product development through PLM. *Manufacturing Engineering*, 144(6).
- Zheng, L. Y., McMahon, C. A., Li, L., Ding, L., & Jamshidi, J. (2008). Key characteristics management in product lifecycle management: A survey of methodologies and practices.
  Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 222(8), 989-1008.
- Zimmermann, J. U., Haasis, S., & Van Houten, F. J. A. M. (2002). ULEO Universal linking of engineering objects. *CIRP Annals Manufacturing Technology*, *51*(1), 99-102.

## APPENDIX A – STEP ONE SIMULATION IN THE PLM SYSTEM WINDCHILL 9.1TM [DESIGN SIDE]

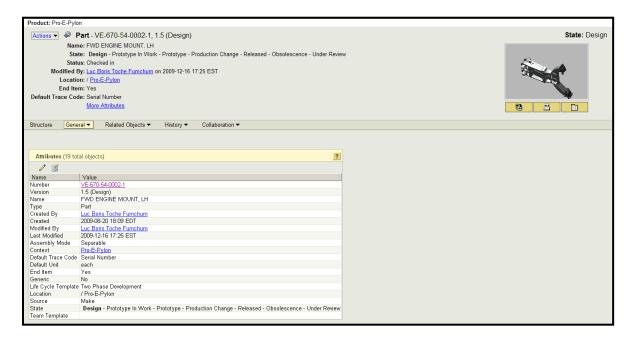


Figure A-1: Fwd Engine Mount general attributes

Figure A.1 shows the general attributes of the forward engine mount as displayed in the component's main page. Anyone who has been granted access to *Pro-E-Pylon* product, as a member of its development, can see this main page and those of all other components. The steps to reach this page are as follow:

- Login into Windchill 9.1 TM.
- Browse the product list.
- Select *Pro-E-Pylon*.
- Browse the loaded structure, locate the end-item VE-670-54-0002-1 and select it.
- Activate *Attributes* in the tab *General*.

Most of the attributes that are then displayed are usually seen on the top of the page and remain there whatever tab is activated in the bottom. Some examples of critical attributes that are quickly available to the user are:

- The part number which is a unique identifier.
- The state of the development; design, prototype in work, released, etc.

- The status of the part which notifies whether it is checked out by someone else or available for modification.
- The last user who modified it.
- The trace code which informs if the part is a configurable item, and how it is traced.

Structure produit								
	Ajouter Enregistrer Promotion Déplacer I	<b>∳</b> Développ		Masquer les ls documents	Afficher les Masqu remplacements remplace	erles Afficherles Nouvelle		Nouvelle Réinitialiser présentation
	Numéro		Actions		Version	Nom	Etat	Quantité
	₩ <u>VE-670-54-0002-1</u>	1	Actions ▼		1.5 (Design)	FWD ENGINE MOUNT, LH	Conception	
	₩ <u>VE-670-54-0013-1</u>	(i)	Actions ▼		1.3 (Design)	YOKE, FWD E.M., LH	Conception	1 unité(s)
	© √E-670-54-0014-1	1	Actions ▼		1.3 (Design)	MOUNT LINK, FWD E.M.	Conception	1 unité(s)
	₩ VE-670-54-0015-1	(i)	Actions ▼		1.3 (Design)	UPPER MOUNT PAD, FWD E.M.	Conception	1 unité(s)
	© √E-670-54-0016-1	<b>(i)</b>	Actions ▼		1.3 (Design)	LOWER MOUNT PAD, FWD E.M.	Conception	1 unité(s)
	▼ <sup>②</sup> <u>∨E-670-54-0095-1</u>	(i)	Actions ▼		1.4 (Design)	FRAME HARDWARE, FWD E.M., LH	Conception	1 unité(s)
	- ∰ AS222374	<b>(i)</b>	Actions ▼		1.3 (Design)	BOLT, DOUBLE HEX, CRES	Conception	2 unité(s)
	□ NAS1149C2232R	1	Actions ▼		1.3 (Design)	WASHER, FLAT	Conception	4 unité(s)
	₩ NAS1291C22	1	Actions ▼		1.3 (Design)	NUT, HEX, SELF-LOCKING	Conception	2 unité(s)
	▼ <sup>③</sup> <u>∨E-670-54-0096-1</u>	(i)	Actions ▼		1.4 (Design)	FIRE WALL HARDWARE, FWD E.M., LH	Conception	1 unité(s)
	₩ AS22552	<b>(i)</b>	Actions ▼		1.3 (Design)	BOLT, DOUBLE HEX, CRES	Conception	6 unité(s)
	₩ NAS1149C0432R	1	Actions ▼		1.4 (Design)	WASHER, FLAT	Conception	12 unité(s)
	₩ NAS1291C04	<b>(i)</b>	Actions ▼		1.4 (Design)	NUT, HEX, SELF-LOCKING	Conception	6 unité(s)
		1	Actions ▼		1.4 (Design)	E.M. HARDWARE, FWD E.M., LH	Conception	1 unité(s)
	₩ AS22950	1	Actions ▼		1.3 (Design)	BOLT, DOUBLE HEX, CRES	Conception	1 unité(s)
	₩ AS221146	(i)	Actions ▼		1.3 (Design)	BOLT, DOUBLE HEX, CRES	Conception	1 unité(s)
	© AS221350	(i)	Actions ▼		1.3 (Design)	BOLT, DOUBLE HEX, CRES	Conception	1 unité(s)
	₩ <u>MS21554-8</u>	(i)	Actions ▼		1.3 (Design)	BEARING, PLAIN, SELF-ALIGNING	Conception	1 unité(s)
	₩ MS21554-10	<b>(i)</b>	Actions ▼		1.3 (Design)	BEARING, PLAIN, SELF-ALIGNING	Conception	1 unité(s)
	₩ MS21554-12	(i)	Actions ▼		1.3 (Design)	BEARING, PLAIN, SELF-ALIGNING	Conception	1 unité(s)
	₩ NAS1149C0832R	<b>(i)</b>	Actions ▼		1.3 (Design)	WASHER, FLAT	Conception	2 unité(s)
	₩ NAS1149C1032R	(i)	Actions ▼		1.3 (Design)	WASHER, FLAT	Conception	2 unité(s)
	₩ NAS1149C1232R	(i)	Actions ▼		1.3 (Design)	WASHER, FLAT	Conception	2 unité(s)
	₩ NAS1291 C08	<b>(i)</b>	Actions ▼		1.3 (Design)	NUT, HEX, SELF-LOCKING	Conception	1 unité(s)
	₩ NAS1291C10	(i)	Actions ▼		1.3 (Design)	NUT, HEX, SELF-LOCKING	Conception	1 unité(s)
	□ NAS1291C12	(i)	Actions ▼		1.3 (Design)	NUT, HEX, SELF-LOCKING	Conception	1 unité(s)

Figure A-2: Fwd Engine Mount eBOM

Figure A.2 shows the forward engine mount product structure as displayed when the corresponding tab is activated. The top and low level parts are all accessible. Their name, version, state and quantity are quickly identifiable. The documents attached to the parts are hidden on the figure. Documents are displayed by selecting the related part using the left hand check row and activating the appropriate icon on the top.

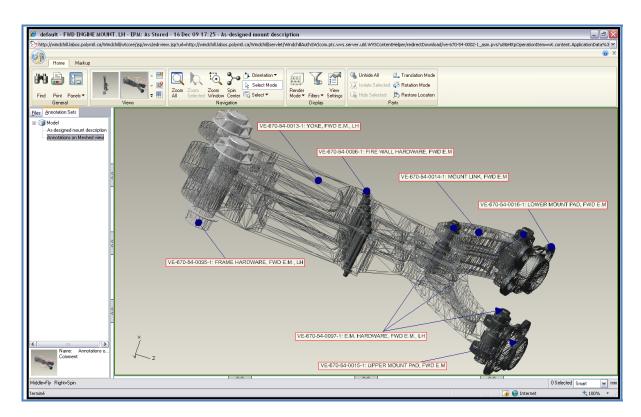


Figure A-3: Annotations for 3D communication

Figure A.3 is an example to show how annotations are used by the product definition team to communicate important information based on 3D representation. A specific pattern to be followed to assemble two components for example could be documented directly in such a view. The file is attached to the eBOM and will further be accessed by the development & test team when designing the test and the mBOM.

The displayed annotation file is only identifying the components with their numbers in *ProductView* <sup>TM</sup> (standard version available in *PDMLink* <sup>TM</sup>). As it will be seen in Appendix B, the eBOM is loaded in the left hand panel of *ProductView* <sup>TM</sup> for *MPMLink* <sup>TM</sup> which is accessed only by the development & test team. A reciprocal selection between the BOM and the 3D mockup is available and this specific annotation file is not really necessary.

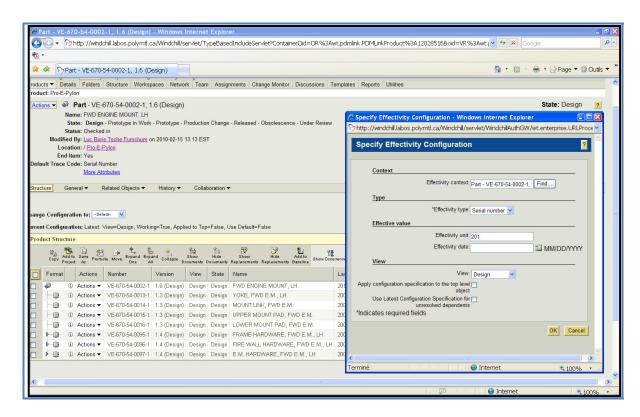


Figure A-4: Effectivity configuration to filter the PS - 1

The test is a structural one and the target here is to filter the BOM to make visible only the structural parts of the assembly that are significant for the intended test. So, even if the *Firewall Hardware VE-670-54-0096-1* is part of the forward engine mount assembly, there is no need to include it during the structural test (see figure A.3). A range of units or in this case, one unit is then configured such as only some parts are effective when this unit is defined in the filter. This allows the development & test team to see only the relevant parts for the test and make its work less cumbersome.

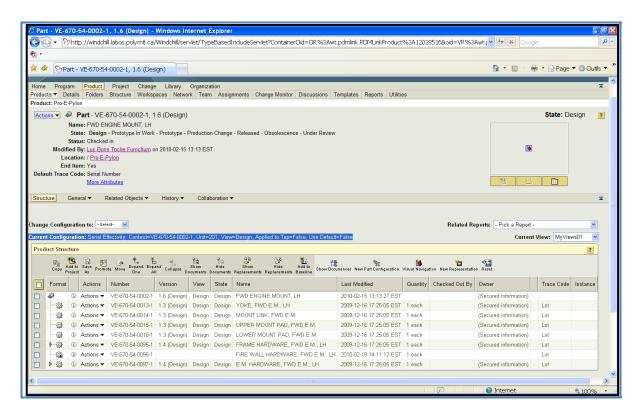


Figure A-5: Effectivity configuration to filter the PS - 2

Figure A.5 shows the result after the filter has been applied. The *Firewall Hardware VE-670-54-0096-1* is not elected as part of the assembly. It carries a specific icon and several descriptions are removed.

It is also possible to see on this figure that the configurable items are already defined. All these parts are traced by lot numbers while the top level end-item *VE-670-54-0002-1* is traced by serial numbers as discussed previously.

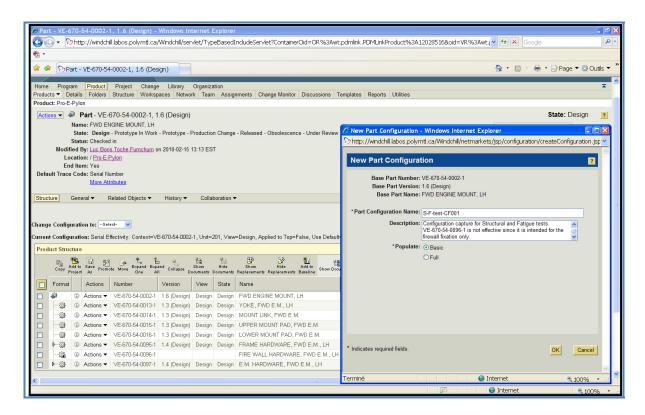


Figure A-6: Configuration methodology – create configuration

The new part configuration wizard is launched via the *Actions* menu on the left of the part's icon. As intended for structural and fatigue tests, a name, *S-F-test-CF001*, and a description are provided for the configuration. The part has reached a certain maturity (1.6) that should be tested and this configuration captures and stores it. Iterations can therefore safely continue. This figure corresponds to the third step of the configuration methodology as described in the diagrammatic analysis in figure 4.6. Idem, a configuration is created for each traceable part to capture it at this state of maturity.

It should be noted that more than one configuration can be created for the same part since different tests can be carried out at different level of maturity.

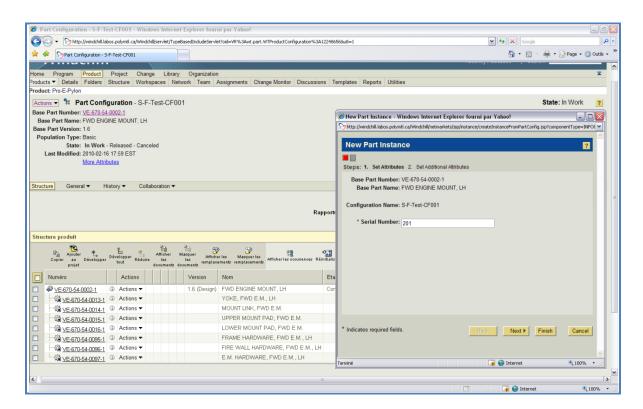


Figure A-7: Configuration methodology – create instance

The forward engine mount part configuration is a whole new object and its information page is displayed in figure A.7. The new part instance wizard is launched via the *Actions* menu on the left of the part configuration's icon. Depending on the trace code, serial or lot, a number is associated to identify the instance which then corresponds to a physical assembled or machined component.

An instance of the forward engine mount with the serial number 201 is created on the figure. As visible in the structure, no low level part instance has yet been elected, or allocated, to constitute and fully define this forward engine mount instance. This is the purpose of the next step.

It should be noted that several instances can be created for the same configuration since more than one physical prototype can be built to track different behaviour of the same configuration.

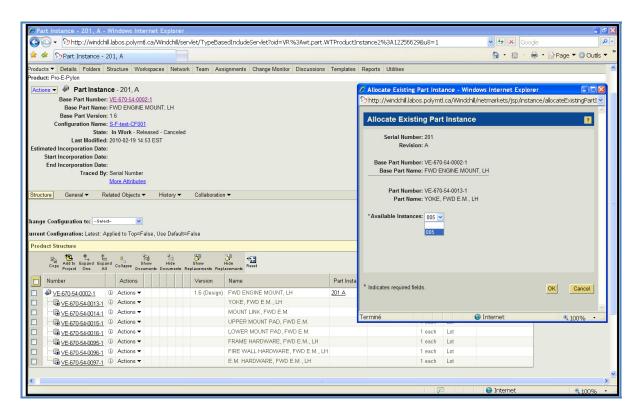


Figure A-8: Configuration methodology – allocate instances

The forward engine mount part instance is a whole new object and its information page is displayed in figure A.8. The part instance allocation wizard is launched via the *Actions* menu on the left of the part instance's icon. As instances of all low level configurable items have already been created, the allocation of an existing instance of the *yoke* is displayed in the figure. This is to say the creation of a low level instance can also be done directly during its allocation.

The allocation displayed on figure A.8 means that the lot number 005 of the *yoke* has been chosen to form part of the physical forward engine mount 201.

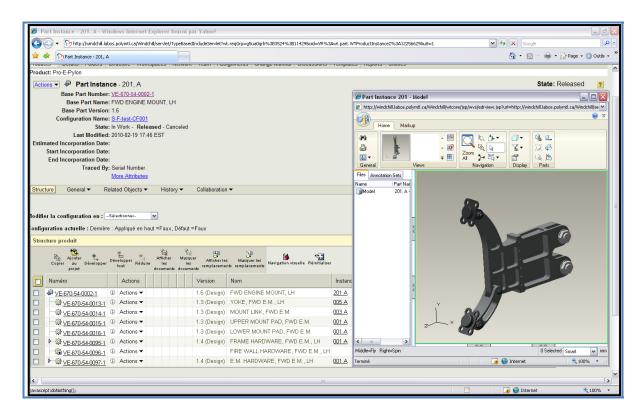


Figure A-9: Configuration methodology – Physical forward engine mount 201

The physical instance of the forward engine mount is completely defined when all the traceable parts have been allocated. Figure A.9 displays the resulting instance 201.

It is possible to see that the *Firewall Hardware VE-670-54-0096-1* is absent and the elected instances of the low level parts for this serial number 201 of the front mount are:

- Lot 005 of the *yoke VE-670-54-0013-1*.
- Lot 003 of the mount link VE-670-54-0014-1.
- Lot 001 of the *upper mount pad VE-670-54-0015-1*.
- Lot 001 of the *lower mount pad VE-670-54-0016-1*.
- Lot 001 of the *frame hardware VE-670-54-0095-1*.
- Lot 001 of the *E.M hardware VE-670-54-0097-1*.

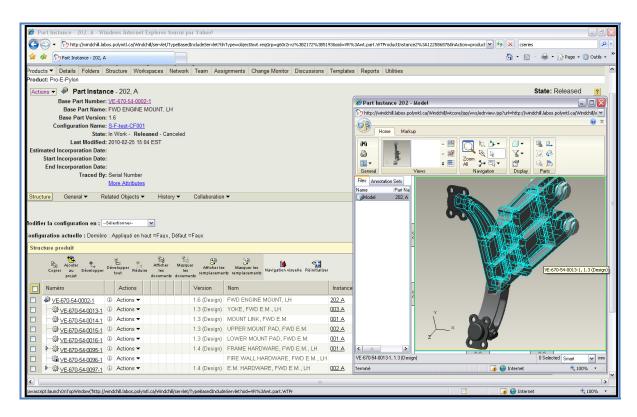


Figure A-10: Configuration methodology – Physical forward engine mount 202

Figure A.10 displays a second physical prototype of the front mount built from the same configuration *S-F-test-CF001* as follow:

- Lot 003 of the *yoke VE-670-54-0013-1*.
- Lot 001 of the mount link VE-670-54-0014-1.
- Lot 002 of the *upper mount pad VE-670-54-0015-1*.
- Lot 001 of the *lower mount pad VE-670-54-0016-1*.
- Lot 001 of the *frame hardware VE-670-54-0095-1*.
- Lot 002 of the *E.M hardware VE-670-54-0097-1*.

The configuration methodology therefore provides the correspondence to the physical prototypes that have been built and also track them during and after the development program.

## APPENDIX B – STEP ONE SIMULATION IN THE PLM SYSTEM WINDCHILL 9.1TM [DEVELOPMENT & TESTING SIDE]

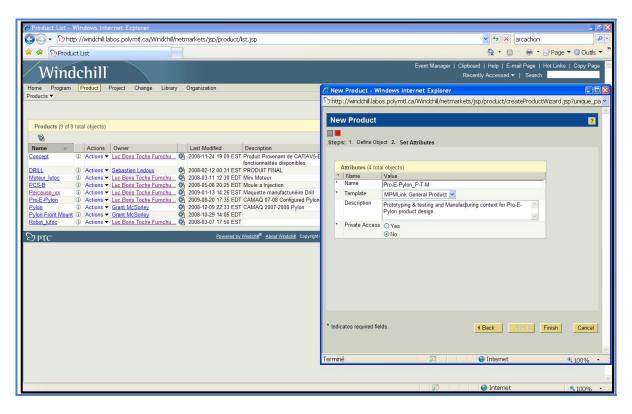


Figure B-1: Create the manufacturing product

A product in Windchill 9.1 TM is basically a container for all the objects created following an activity which is either a design activity or a manufacturing one. The reason why a manufacturing product is created when the baseline design product already exists is mainly to separate the contexts in which the two stakeholders work. The information packages are then classified in a more structured and rational way. The other important reason of creating a manufacturing product is that any BOM, such as the mBOM, can only be stored in a product and a product cannot contain two BOMs. Moreover, the manufacturing product is generated from the MPMLink General Product template, which then allows the management of the related objects as part of the MPM module.

To create a manufacturing product, the steps below have to be followed:

- Login into Windchill 9.1 TM.
- Browse the product list.

- Hit the *New product* icon to launch the wizard.
- Write down a name, select MPMLink General Product template and define the access security.

Pro-E-Pylon product for prototyping, testing and manufacturing, *Pro-E-Pylon\_P-T-M*, is defined as such.

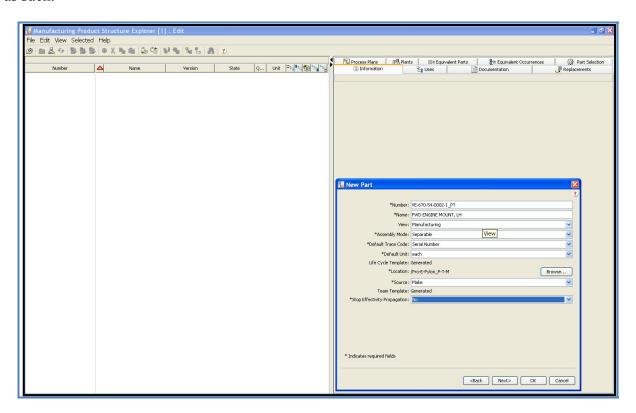


Figure B-2: Create the mBOM top level item

The mBOM is built in the *Manufacturing Product Structure Explorer* (Manufacturing PSE). This explorer is launched from the product's main information page; it is the second in the row of the explorers' icons. Once the explorer is set up, the *new part* wizard is started from the *File* menu. As seen in Figure B.2, the default context is *Pro-E-Pylon\_P-T-M* product. Key attributes are defined for this manufacturing part, these are:

- The part number, which is enabled only for parts created for manufacturing reasons or/and which have no strict equivalent in the eBOM.
- The view, which identifies the type of the object; a design object or a manufacturing one.
- The assembly mode; separable, inseparable, closed box component, etc.
- The trace code which informs if the part is a configurable item, and how it is traced.

- The default unit to quantify the item.
- The source to know if the item is fully made, outsourced, etc.

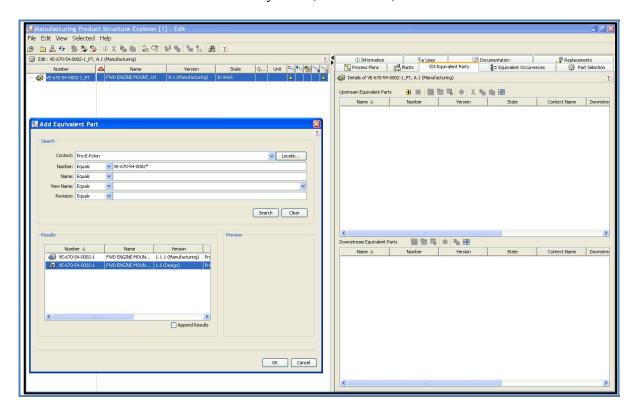


Figure B-3: Define the equivalence link

Once the mBOM top level item VE-670-54-0002-1\_PT has been created, the equivalence to the upstream design part has to be enabled to follow the complementary structure approach. There is no need for this operation when it comes to the low level equivalent parts since their manufacturing views will be directly attached to the mBOM and the links will be enabled automatically. The link is enabled manually only for the top level item since it is a manufacturing assembly with a structure to be defined. This assembly cannot be the straight manufacturing view of the top level eBOM item, which barely results in the same structure.

The links to the eBOM are enabled by following the steps below:

- Activate the *Equivalent parts* tab in the right hand panel.
- Hit the *Add a new part* icon.
- Use appropriate criteria to locate the equivalent eBOM part.
- Follow the wizard and enable the occurrence link as displayed in figure B.4

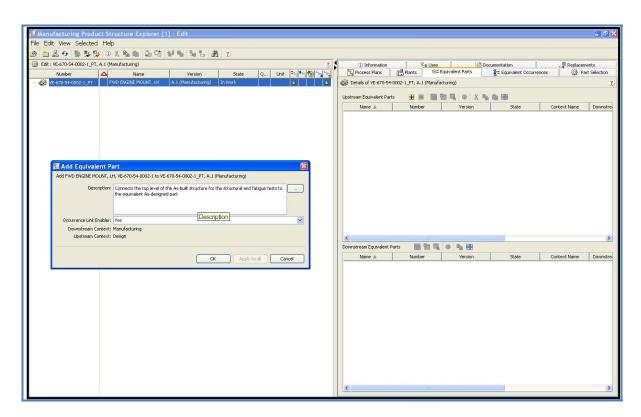


Figure B-4: Define the occurrence link

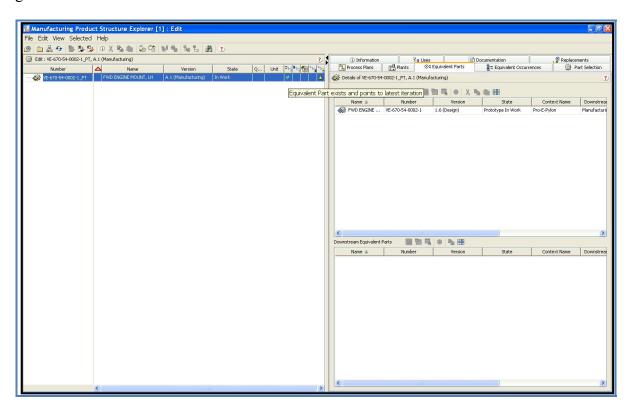


Figure B-5: mBOM – eBOM links - 1

When the links are enabled, the equivalent parts are listed in the tab and green marks are displayed next to the part in the left hand panel (see figure B.5). These attest that the equivalent part exists and points to the latest iteration, but also, in the occurrence check row, that position is calculated (see figure B.6).

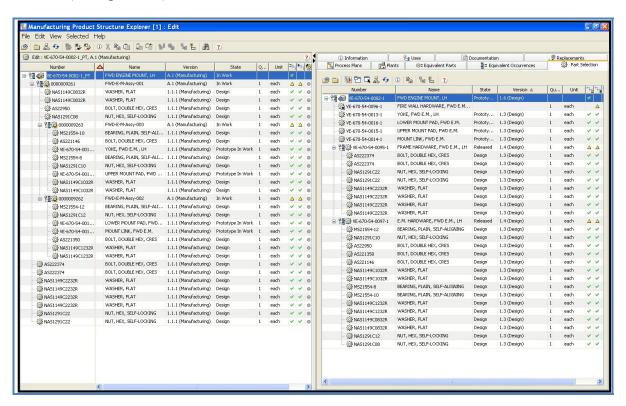


Figure B-6: mBOM – eBOM links – 2

In case there is a modification on the Design side, a clock icon replaces the green mark and an assessment of the modification is done to eventually update the links.

The manufacturing parts created following the prototyping, assembly and test strategy are quickly identifiable. They bear the yellow triangle icon in the upstream equivalence and occurrence rows. This means that no equivalent eBOM parts exist.

Idem, in the right hand panel, where the eBOM is loaded in the *Part selection* tab, the yellow icons indicate the parts that have no downstream equivalent.

Finally, it is possible to see through the mBOM how the prototyping and manufacturing strategies have driven a new arrangement of the eBOM parts that is suitable for the assemblage.

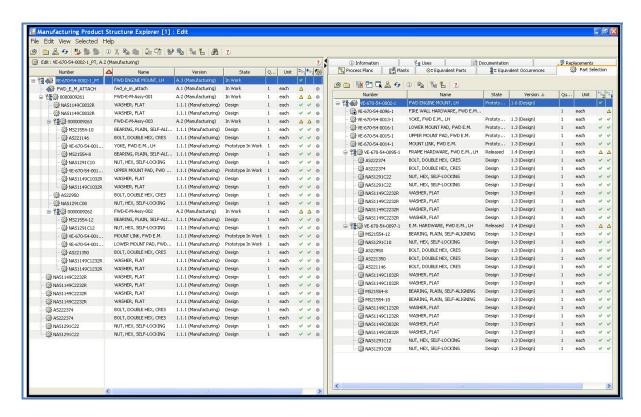


Figure B-7: Add a new component - 1

In figure B.7 a new a component with its CAD document is added for the test. The component Fwd\_E\_M\_Attach does not exist in the eBOM and it is one that the 3D representation is not retrieved from the eBOM. The component is created in *PDMLink* <sup>TM</sup>, attached via the manufacturing PES and positioned in *ProductView* <sup>TM</sup> for *MPMLink* <sup>TM</sup>. Figure B.8 displays the result in an annotation file.

All other instrumentation components are added by following the same pattern.

The yellow triangle icons remain to indicate that no equivalent parts exist in the eBOM.

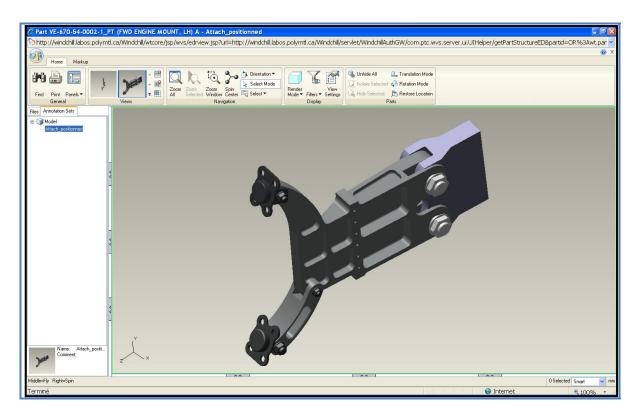


Figure B-8: Add a new component - 2

It can be noticed that this mock-up as assembled is only visible by the testing team.

Being able to add a component and its CAD document to the mBOM and make it visible in the 3D mock-up retrieved from the eBOM offer a new playground to the testing team to design the test. Multiple scenarios with diverse components could be assessed and validated on a 3D basis.

Moreover, if an eBOM part needs to be modified for a sensor to go through it for example; this could be done by adding the modified part as a new component instead of modifying the original part or trying to find a way to add the test team's modified part as another version in the eBOM.

After adding this modified eBOM part in the mBOM, a reference link should be enabled with the original part to identify to what the modified part is equivalent and track any eventual Design modification.

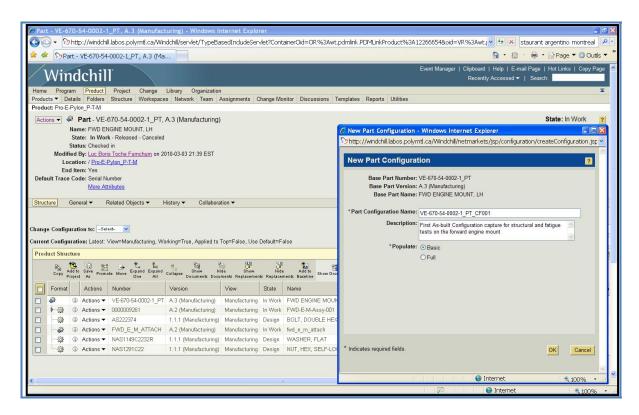


Figure B-9: Configuration methodology – create configuration

The same pattern as in figure A.6 is followed to create the manufacturing part configuration. The new part configuration wizard is launched via the *Actions* menu on the left of the part's icon. A name, *VE-670-54-0002-1 PT-CF001*, and a description are provided for the configuration.

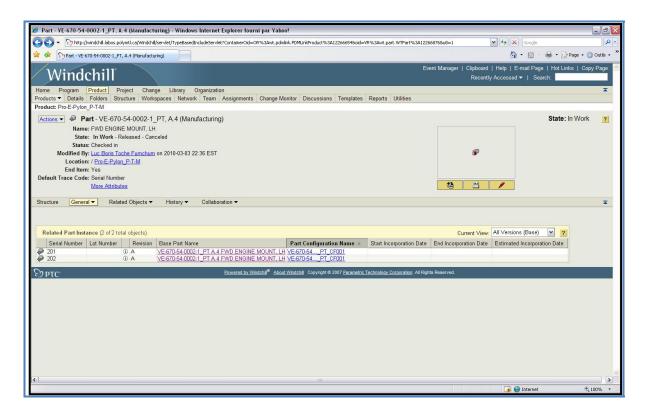


Figure B-10: Configuration methodology – part instance

After following the configuration methodology as previously described, the testing team is also able to make the correspondence between the allocated manufacturing structures and the physical prototypes identified with their serial numbers 201 and 202.

The testing team is not tracing the hardware components *frame hardware VE-670-54-0095-1* and *E.M hardware VE-670-54-0097-1* since it has been found that they do not have a significant influence on the structural test. The added component is untraced as well as the parts added following the manufacturing strategy.

Figure B.11 and B.12 displays the manufacturing instances corresponding to the physical front mounts 201 and 202. The allocated low level parts match the design instances for each serial number. This ensures that the two teams have defined and mirror the same physical prototypes as exhibited in the diagrammatic analysis in figure 4.6.

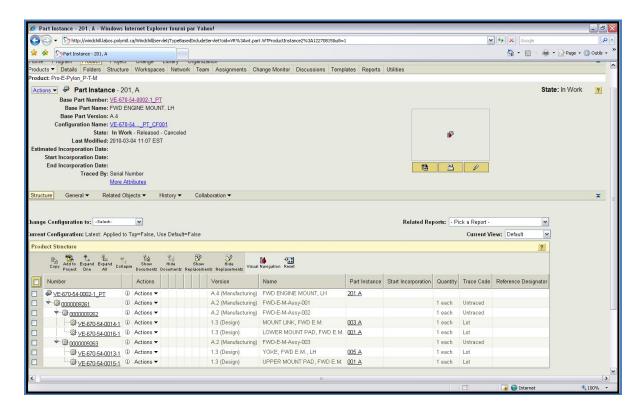


Figure B-11: Configuration methodology – manufacturing part instance 201

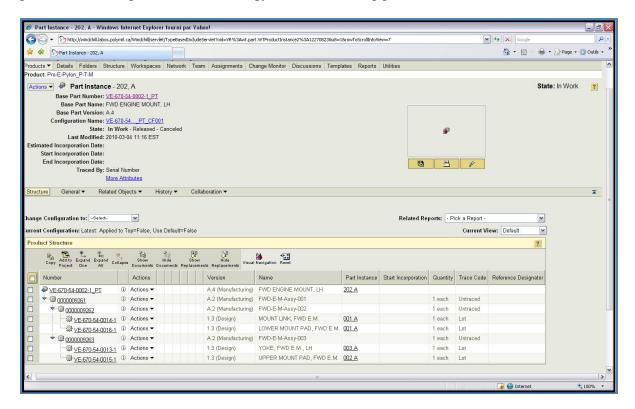


Figure B-12: Configuration methodology – manufacturing part instance 202

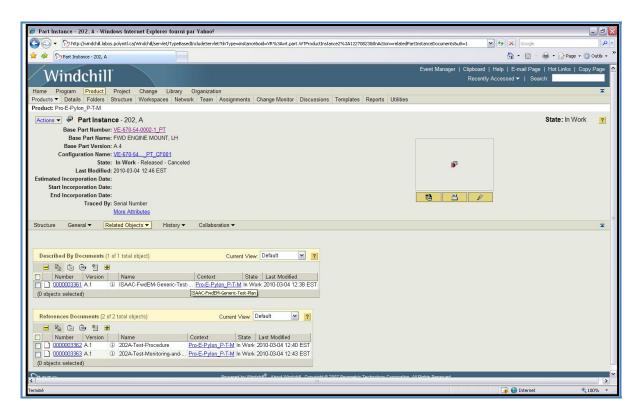


Figure B-13: Add prototype and test related documents

The final step is to document each test and the results and therefore to attach all the relevant documents to the related prototype instance. Figure B.13 displays the generic front mount test plan attached to the instance 202 as a description document. The test procedure and the test results specific to this front mount prototype 202 are also attached as references documents. This information is available for the whole team when browsing tracking and analyzing the prototypes and tests.

## **DISCLAIMER**

Some commercial software systems are identified herein in order to facilitate understanding. Such identification does not imply that the software systems are necessarily the best available for the purpose.