



HAL
open science

Identité, évolution et discernabilité des objets dans les systèmes d'information et les bases de données

Gilbert Giacomoni, Jean-Claude Sardas

► **To cite this version:**

Gilbert Giacomoni, Jean-Claude Sardas. Identité, évolution et discernabilité des objets dans les systèmes d'information et les bases de données. 2012. hal-00753115

HAL Id: hal-00753115

<https://minesparis-psl.hal.science/hal-00753115v1>

Preprint submitted on 20 Nov 2012

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

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

Identité, évolution et discernabilité des objets dans les systèmes d'information et les bases de données

Gilbert Giacomoni

Institut de Recherche en Gestion - Université Paris 12 (UPEC)
61 Avenue de Général de Gaulle 94010 Créteil
gilbert.giacomoni@u-pec.fr

Centre de Gestion Scientifique □ Chaire TMCI (FIMMM) - Mines ParisTech
60 Boulevard Saint-Michel 75006 Paris
gilbert.giacomoni@mines-paristech.fr

&

Jean-Claude Sardas

Centre de Gestion Scientifique - Mines ParisTech
60 Boulevard Saint-Michel 75006 Paris
sardas@mines-paristech.fr

RÉSUMÉ: Les principes et les modalités de gestion des évolutions des données techniques proposées notamment par les Product Lifecycle Management (P.L.M) sont étudiés dans les contextes de forte évolutivité des objets techniques et de reproduction en série. Les tentatives de conjugaison des fonctionnalités offertes par les P.L.M. conduisent à des choix inconciliables avec un traitement entièrement automatisé, et les résultats escomptés doivent donc être relativisés. Les questionnements théoriques renvoient aux fondements de l'identité et de la discernabilité, de l'unique et du multiple, des évolutions et des invariances, qui sont explorés. Une approche formelle de la dénomination des objets est proposée, dans la continuité de l'algèbre d'E.F. Codd, pour relayer les stratégies de gestion des relations. L'interchangeabilité des objets et des relations, des compositions et des contextes d'emplois, est redéfinie de manière relative et conditionnelle. Les stratégies de gestion des évolutions de configurations sont optimisées selon deux principes génériques visant à préserver les invariances ou à en concevoir de nouvelles sur des bases constructibles. Ces principes entrent en action selon la manière dont les situations industrielles conjuguent les régimes de reconception (innovation) et de reproduction (sériation). Les propositions ont été nourries par des travaux pluriannuels conduits en milieu industriel (aéronautique notamment) ainsi que dans la santé, et ont eu des implications pilotes en entreprise. Les principales solutions P.L.M. du marché ont été examinées.

Mots clés: PLM, interchangeabilité, configuration, identité, discernabilité, évolutions, invariances,

Identity, Evolution and Discernibility of Objects in Information Systems and Databases

ABSTRACT: The principles and the modalities proposed, in particular by Product Lifecycle Management (P.L.M.) to manage the technical data evolutions, are studied in the contexts of intensive renewal and creation of technical objects (innovation) with serial reproduction (serialization). The attempts to conjugate the various functionalities offered by P.L.M. lead in fact to irreconcilable choices with a fully automated process, and the expected results must be relativized thereof. The questions refer to theoretical foundations of identity and discernibility, of the single and multiple, evolutions and invariances, which are explored. A formal approach to object naming is proposed, in line with the algebra of E.F. Codd^[3], to relay strategies based on relationship management only. The interchangeability of objects and relations, of compositions and contexts of use, is redefined as being relative and conditional. Strategies to manage evolutions of configurations are optimized using two generic principles aimed at preserving invariances or designing new constructible ones. These principles come into play depending on how industrial situations combine design (innovation) and production (series) rates. These multi-annual investigations were held in industrial environment (in particular aeronautics) and in healthcare with experimental applications. The main P.L.M. solutions of the market have been examined.

Keywords: PLM, interchangeability, configuration, identity, discernibility, evolutions, invariances,

Introduction

Les entreprises doivent gérer des produits, des installations ou des projets de nature profondément complexe et évolutive. Ce qui représente des milliers de références de définition et plus encore d'articles en exploitation, avec des versions successives et des documents qui leur sont associés tout au long de leur vie. Pour assurer la fiabilité des définitions, des exploitations, des maintenances opérationnelles, il faut être en mesure de garantir la cohérence et la gestion de la qualité de l'ensemble des données techniques: suivi des événements, des modifications nécessaires, des impacts possibles sur l'ensemble, des non-conformités, des actions, etc. Ces exigences se renforcent dans une économie de la variété où les produits se diversifient et se renouvellent à un rythme accéléré. Pour améliorer leur compétitivité, réduire les cycles et les coûts, accroître la réactivité, maîtriser la complexification de leurs produits et de leurs procédés, les entreprises doivent s'informatiser, en reliant, progressivement ou simultanément, tous les segments fonctionnels: R&D, ingénierie, marketing, qualité, achats, etc. Nombre d'entre elles se tournent notamment vers des solutions de type *Product Lifecycle Management* (PLM). Il s'agit d'outils d'entreprise, des progiciels, à la fois modulaires et intégrés (Mostefai et Batouche, 2005) autour d'un noyau unique¹, permettant de gérer le cycle de vie des produits (Batenburg et al., 2005; Pol et al., 2005), de leur conception à leur mise au rebut (Stark et al., 2004; Amann, 2002). En somme, un ensemble logiciel modulaire et intégré, traitant et partageant entre les différents acteurs, toutes les données de l'entreprise représentant les produits et les processus liés à l'innovation et à l'évolution. Pour mettre en œuvre de telles instrumentations et engranger rapidement les bénéfices escomptés, les entreprises doivent consentir des efforts et des investissements de grande ampleur. Il nous a paru important aujourd'hui, de nous pencher sur ces solutions pour examiner les principes et les modalités des options de gestion des évolutions des données techniques proposées et voir en quoi ces options répondent aux besoins et permettent ou non de maîtriser ces évolutions, et ce en particulier dans le cas difficile où les évolutions se traduisent par des impacts multiples sur les configurations. La clé de la gestion des données techniques est en effet, selon les promoteurs des solutions *PLM*, cette gestion des configurations, qui doit permettre d'avoir, à tout moment, une connaissance exhaustive des données techniques afférentes à un programme, un projet, un produit.

L'exemple ci-après emprunté à l'aéronautique (source Rome AFP), nous donne un aperçu des enjeux. Le 6 août 2005, au large des côtes siciliennes, un avion ATR²-72 tente un amerrissage tragique. Ses deux moteurs se sont arrêtés quelques minutes plus tôt. La cause principale de l'accident est une erreur d'installation d'une jauge de carburant. L'équipage de l'ATR-72 croyait avoir 3.000 litres de kérosène. La veille, un technicien de maintenance avait dû remplacer la jauge et le système informatique lui avait proposé celle d'un ATR42, censée être interchangeable en taille et en connectique. Seule une petite inscription rappelait la capacité de chaque réservoir et celui de l'ATR-42 était d'une capacité deux fois inférieure à celui de l'ATR-72. Les contrôles extérieurs et en cabine n'avaient rien décelé de suspect par rapport aux marges tolérées par le constructeur. Erreur informatique ou erreur humaine? L'interchangeabilité n'avait pas une valeur totale mais restreinte (valable sous certaines conditions non précisées en informatique). Nous aurions pu tourner notre regard vers l'industrie logicielle³ ou pharmaceutique⁴, les problématiques auraient été analogues.

1 Au sens informatique, un noyau unique offre une structure alternative aux systèmes hétérogènes interfaçant des solutions logicielles ayant leur propre noyau et spécifiques à chaque domaine fonctionnel [conception, production, finances...]

2 Filiale d'EADS, leader mondial des avions à turbopropulseur.

3 A titre d'exemple, l'explosion de la première fusée Ariane V à cause (selon le rapport de la commission d'enquête) du Système de Référence Inertielle (SRI), logiciel repris tel quel d'Ariane IV (interchangeabilité supposée totale). Ariane V, avec des moteurs plus puissants, s'est inclinée plus rapidement qu'Ariane IV pour récupérer l'accélération liée à la rotation de la Terre. Le logiciel SRI a considéré cette inclinaison d'Ariane V non conforme au plan de tir (d'Ariane IV) et a provoqué l'ordre de procéder à une correction importante de trajectoire par rapport à une déviation qui, en fait, ne s'était pas produite.

4 Le générique partage avec l'original la même substance pharmacologique active (principe actif), la même forme galénique, la même voie d'administration, le même dosage et les mêmes indications (recherche clinique; autorités sanitaires) que le médicament dont il est la copie. Il est interchangeable avec la préparation originale et est équivalent sur le plan thérapeutique. Mais la preuve est complexe et les études cliniques sont établies indirectement, par bioéquivalence. C'est également le cas pour des médicaments originaux lorsqu'il y a une modification de leur formulation, changement de processus ou de site de fabrication.

Pour gérer les évolutions de configuration, tout un panel de fonctionnalités est prévu dans les *PLM*, avec une gestion des évolutions de configuration par dates ou par rangs d'applicabilité⁵, des mises à jour tenant compte des arborescences de codifications normalisées, des cas d'emploi, des interchangeabilités, etc. Il n'y aurait a priori plus qu'à sélectionner les fonctionnalités nécessaires et suffisantes parmi celles proposées, pour gérer correctement les données techniques d'une activité industrielle donnée. Cet exercice n'est pas si évident (Eicher et al., 1984; Hatchuel, Sardas et Weil, 1988). Les fonctionnalités sous-tendent des formalismes gestionnaires (pour productions en série, ou plutôt unitaire, avec ou sans assemblage, etc.) dont elles sont issues. L'existence de ces fonctionnalités résulte en effet d'une traduction logicielle de pratiques industrielles réelles (observées) ou imaginaires (espérées) ayant servi de modèle à leur conception. Personnaliser l'usage d'outils P.L.M ainsi conçus en transcrivant de manière pertinente l'activité désirée suppose de savoir choisir le(s) bon(s) formalisme(s) puis de modéliser l'activité dans ce(s) formalisme(s), quitte à adapter certaines pratiques pour les rendre conformes au(x) formalisme(s) ou, si c'est possible, adapter le(s) formalisme(s). C'est toute la sensibilité des rapports entre les systèmes d'information, les individus et les organisations (Marciniak et Rowe, 2008). Comment donc, le mode d'emploi d'un outil à géométrie suffisamment variable pour s'adapter à des activités industrielles très différentes, prévoit-il de transcrire des activités hybrides conjuguant innovation intensive et reproduction en série (sériation) avec fréquemment des impacts multiples sur les configurations?

Nous nous sommes donc mis en quête de comprendre la logique des formalismes sous-tendus par les fonctionnalités proposées dans les PLM et comment ils pouvaient se combiner pour certaines activités hybrides, partant de l'idée qu'une entreprise peut, pour tout ou partie de son activité, à un moment ou à un autre de son évolution, devoir conjuguer innovation intensive et sériation. Concernant les activités hybrides, il ressort de nos travaux, que la problématique de la gestion de vagues de modifications ayant des impacts sur les configurations, n'a pas été résolue sur un plan conceptuel et pratique. Les tentatives de conjugaison des fonctionnalités offertes par les PLM conduisent en fait à des choix inconciliables avec un traitement automatisé, et les résultats escomptés doivent donc être prudemment relativisés. Ce sera tout l'objet de la première partie. En conséquence, nous nous sommes efforcés de définir les conditions et les principes d'une gestion automatisable des activités hybrides, notamment via une notion d'interchangeabilité relative, conditionnelle, et une conception nouvelle de l'identité et de la dénomination des objets permettant de lier, de manière automatisable, leur composition et leurs configurations d'emplois. Nous en explorerons les aspects théoriques et aussi historiques, puis ébaucherons in fine une typologie des formalismes possibles de gestion des évolutions de configuration en fonction des situations industrielles génériques conjuguant de manière variable innovation et sériation. Ce sera l'objet de la seconde partie.

1. PLM et gestion des évolutions de configurations : problématiques industrielles et questionnements théoriques

Les outils *Product Lifecycle Management* (PLM), s'appuient sur des technologies avancées qui ont émergé dans l'univers des projets aéronautiques et automobiles avant de se diffuser à d'autres secteurs industriels⁶: «La technologie PLM trouve son origine dans les systèmes de gestion des données techniques (...) initialement adoptée dans l'aéronautique et l'automobile, (elle) s'est progressivement diffusée dans d'autres industries plus traditionnelles au début des années 2000 avec des solutions développées par des éditeurs comme Dassault Systèmes, Siemens ou PTC»

⁵ Le terme d'effectivité est aussi employé pour exprimer la différence entre applicable et appliqué. Effectivité exprime la qualité de l'adéquation entre ce que l'on fait effectivement et ce que l'on voulait faire.

⁶ Industrie discrète et continue

(Merminod, Mothe, Rowe, 2009). Les PLM proposent un environnement harmonisé⁷ pour tous les acteurs et une base de données unique pour modéliser⁸ les objets, les processus⁹ (Grieves, 2006) et les connaissances (Benbya et Meissonier, 2007). Il est possible de gérer des articles (terme générique pour désigner une matière première, un composant, un sous-ensemble, un produit fini), des nomenclatures (structure arborescente des produits), des dossiers (de conception, de fabrication, etc.), des modifications et leur applicabilité (effectivité), des configurations (variantes, options, substitution, etc.), rechercher des cas d'emploi (ensemble des produits dans lesquels est montée une pièce), réutiliser des données existantes au sein d'autres produits, etc.

Dans certains domaines ou secteurs¹⁰, l'évaluation des PLM a été positive tant sur le plan de la productivité que sur celui de la fiabilisation du processus de développement. Mais qu'en est-il des entreprises dont la survie dépend d'une dynamique d'innovation intensive génératrice d'objets multifformes et en même temps d'une reproduction sériée de tels objets? Un objet multifformes est un objet technique pouvant exister sous plusieurs formes qui ne sont pas différenciables par des dénominations distinctes, et qui ne sont pas pour autant interchangeable. Il ne s'agit pas de multivues (multi-view ou viewpoint) qui est un autre concept représentant un objet technique pouvant être exploité dans différents contextes¹¹ liés à différents points de vue¹² associés aux acteurs métiers impliqués dans le cycle de vie (Bernard, 1996). Le concept de multivues a fait l'objet de travaux de recherche dans les domaines du PLM et du développement des produits (Gomes et Sagot, 2002; Bronsvoort et Noort, 2004; Bouikni et al., 2008; Noel, 2006; IBM et Dassault Systèmes, 2008). La construction, l'évolution de chaque vue, et le contrôle de la cohérence dans le modèle produit, font partie des enjeux toujours en vigueur [Bernard & Perry, 2003] car la complexité combinatoire des relations possibles soulève le problème de l'algèbre relationnel et de la sémantique des identités. Le multifformes est un concept d'objet qui émerge des environnements fortement évolutifs où les délais impartis ne permettent plus, ni d'effectuer une analyse combinatoire des risques pouvant garantir une totale équivalence d'emploi, ni de distinguer tous les cas de figures avec des avalanches cascades dans les bases de données et les documentations associées. Ce concept de multifformes permet aux concepteurs de gérer les configurations en brassant les contenus et les contextes suivant un principe d'identité relative, en somme paradoxal du point de vue de l'identité absolue classique.

Le mode d'emploi d'un PLM prévoit-il un usage adapté à ce genre d'objet? Il nous faut, pour tenter de répondre à cette question, commencer par comprendre (cf. 1.1 et 1.2) à la fois les différents usages possibles prévus dans les PLM et aussi comment naît cet objet paradoxal qu'est un multifformes (cf 1.3).

1.1. Faire l'appoint des modifications pour l'exemplaire à produire

Partons de la description la plus simple possible d'un objet technique (Simondon, 1958): un ensemble d'éléments liés entre eux, globalement solidaires dans leur fonctionnement. Cela se matérialise par une liste des éléments entrant dans la composition de l'objet et par des indications sur la façon de réaliser l'objet. La documentation accompagnant des produits livrés en kit à monter soi-même donne une bonne illustration de ce cas de figure simple. Supposons, à un instant initial, que la définition de l'objet (c'est-à-dire sa composition et son mode de réalisation)¹³ soit partagée par l'ensemble des acteurs et tienne lieu de référentiel industriel d'origine. Cette définition ne peut

7 L'ergonomie d'un interface homme-machine unique peut être préférable (pour les formations, les changements d'applications) à celle d'interfaces multiples en environnement hétérogène.

8 Données techniques, spécifications, nomenclatures, gammes, (etc.) et séquençement des processus.

9 Automatisation de certaines tâches de conception ou de production

10 Par exemple le cas de l'électroménager au sein du Groupe Seb Moulinex (Merminod, Mothe, Rowe, 2009)

11 Au sens contexte de conception [Rehman & Yan, 2007] et configuration produit

12 Structurelles, technologiques, géométriques, fonctionnelles, comportementales ou contextuelles

13 La composition d'un objet limite la connexité des relations possibles entre objets (de type parents-enfants uniquement et que la plupart des systèmes d'information supporte). Mais l'incomplétude de l'algèbre des concepteurs vaut quelle que soit la connexité initiale possible.

rester figée dans le temps, en raison d'une part de l'innovation (en particulier l'innovation incrémentale) et d'autre part de la nécessité de corriger les erreurs, de prévenir les risques, et d'améliorer la qualité des produits. Elle va donc évoluer au rythme des modifications apportées. Ces modifications sont instruites et classées en fonction des impacts générés. Une première manière de gérer les états successifs de la définition est d'en mesurer les écarts par rapport à l'original en recensant les modifications survenues. Dans cette logique, l'objet technique est défini avec sa définition d'origine, augmentée des modifications applicables pour des exemplaires donnés. C'est ainsi que peuvent se coordonner les différents intervenants (Giacomoni et Sardas, 2011). A compter d'une date, d'un contrat client, ou d'un rang donné (numéro d'exemplaire de l'objet) généralement négocié(e) en fonction des stocks existants pour limiter les obsolescences, l'ajout des modifications doit être fait. On reconnaît parfaitement cette pratique dans le cas ci-après: « Réglage moteur, modification d'une porte ou d'un câble électrique, chaque semaine, (etc.) une cinquantaine de nouvelles modifications. Minutieuses et rigoureuses, ces opérations d'analyse d'impact sur la documentation applicable, de cas d'emploi, de simulation de résultat et de synthèse s'exécutent très lentement en comparaison à d'autres industries. Normal lorsque l'on sait que chaque modification (□) équivaut à un changement sur le seuil de tir de Kourou et surtout sur le lanceur. Et vu que ce dernier est estimé à 130 millions d'euros, mieux vaut ne pas se tromper. Avec, fin février 2007, 25 vols commerciaux réussis pour seulement 4 échecs, Ariane 5 vise l'excellence. » (Fodor X., 2008). Une manière simple de comprendre cette logique est de considérer le cas du mode d'emploi d'un produit, prévu pour des consommateurs de différents pays dont les dispositifs légaux respectifs (normes, réglementations, etc.) n'évoluent pas de manière harmonisée. Le mode d'emploi doit toujours respecter les derniers dispositifs légaux en vigueur (formulation, informations, mentions légales, etc.). Supposons que le produit lui-même demeure inchangé et que seul le mode d'emploi doive s'adapter à l'évolution de la réglementation dans un des pays couverts. Un mode d'emploi conçu avec une page par langue (la page 1 en français, la page 2 en russe, la page 3 en chinois, etc.) n'offre guère d'autres choix, dans le cas où, pour un pays donné, la norme applicable évolue à une date donnée, que de rééditer l'intégralité du mode d'emploi qu'il faudra insérer dans les cartons d'emballage à compter de la date prévue, avec la modification n° M à la page n° P. Ainsi, pour contrôler la bonne version du mode d'emploi correspondant à un produit vendu à une date et dans un pays donnés, il est nécessaire de s'assurer que la modification n° M à la page n° P est bien présente par rapport à la version d'origine. Les cas de figures entrant dans ce schéma sont nombreux. On peut prendre l'exemple d'une prescription (correspondant à la définition d'un objet technique) de médicaments (éléments composant cet objet technique) pour un patient hospitalisé. A chaque visite du médecin et en fonction de son état, la prescription peut évoluer. Un médicament peut être remplacé, ou seule sa posologie peut être réajustée. C'est une gestion nominative (nom du patient), chaque fois l'ensemble de la prescription (pour ce malade précis) dans sa dernière version (date de la dernière visite) est prise en référence, pour éviter une erreur de médication croisée. Cette philosophie gestionnaire est essentiellement sécuritaire et convient uniquement pour des productions à l'unité ou en très petites séries. Il suffit d'avoir à l'esprit l'exemple du mode d'emploi et de l'imaginer englobant la description complète de l'équipement (composition, matières, configuration, etc.), pour matérialiser très vite la lourdeur et le coût d'une réédition systématique à la moindre modification de page. C'est de tonnes de documentation qu'il s'agirait pour un équipement industriel. Impossible de reproduire en série dans de telles conditions. Du reste, cette pratique s'accommode mal des contraintes de dépannage. Le remplacement d'un élément solidaire de l'objet global ne peut être considéré isolément et implique la reproduction intégrale de l'objet. Le remplacement d'une page implique le remplacement de l'ensemble du mode d'emploi. Pour spécialiser les productions, accroître la réactivité, assurer le réassort, reproduire en série (sérié) les différents éléments, la condition préalable est donc une définition gérable de manière autonome de chaque élément par rapport à l'ensemble de l'objet. Il faut dans cette optique que la définition de chaque élément précise quelles modifications lui sont appliquées et qu'un ensemble d'éléments

modifiés incorpore exactement les modifications prévues à compter d'un rang (d'une date, etc.) donné. C'est ce que nous allons développer maintenant.

1.2. Renommer ce qui diffère dans la composition reproduite : principe d'interchangeabilité

Par rapport au formalisme précédent, où chaque objet était identifié dans sa globalité, une identification (codification normalisée)¹⁴ unique est cette fois attribuée à chaque élément. Elle est commune avec sa définition¹⁵ et suffisante pour le reproduire. Si nous reprenons notre exemple du mode d'emploi polyglotte, cela revient à concevoir autant de modes d'emploi différents qu'il y a de langues différentes. Chacun aurait sa propre identification et sa reproduction serait autonome. Il est évident qu'en cas d'évolution de la norme d'un pays donné, il suffirait de modifier uniquement le mode d'emploi destiné au pays concerné et de l'identifier différemment pour ne pas confondre les versions successives. Mais une fois le mode d'emploi dans sa bonne version inséré dans le carton d'emballage, comment repérer parmi tous les cartons, ceux destinés aux pays dont la norme a changé? Plusieurs solutions sont envisageables. Les stocks de la version précédente pourraient être épuisés avant que ceux de la nouvelle version ne soient constitués. Mais cette solution n'est pas toujours possible. Par exemple en raison du réassort ou du dépannage. Toutes les versions doivent être disponibles si un client veut remplacer un équipement par un modèle strictement identique ou pour effectuer un échange standard. De plus, une version anglaise n'est pas restreinte au seul pays affecté par une évolution normative. Plusieurs pays peuvent être destinataires de la langue anglaise. Il serait contreproductif de réduire les quantités économiques à lancer en impression, en limitant la durée de vie d'une version juste pour éviter la survivance des versions successives. Une autre solution pourrait consister à identifier différemment les cartons en fonction de la version du mode d'emploi qu'ils contiennent. Cette solution n'est pas toujours possible non plus. Car si le mode d'emploi concerne un équipement qui doit être intégré dans un autre ensemble plus important, jusqu'où va-t-il falloir répercuter la modification (marquages physiques, bases de données, etc.)? De plus que se passe-t-il si les évolutions se démultiplient avec la multitude de normes (recyclage, signalétique, etc.) et directives applicables? On voit bien que la distinction systématique des versions peut atteindre ses limites dès que le scénario se complexifie un peu. Une notion vient pallier aux répercussions en chaîne d'une modification: l'interchangeabilité¹⁶. Par exemple, pour un carton contenant un mode d'emploi de langue anglaise, destiné aux pays anglophones non concernés par l'évolution normative (par exemple tous sauf l'Australie), la version du mode d'emploi est totalement indifférente. Les cartons destinés à ces pays sont interchangeables. Mais il suffira qu'une autre norme évolue à son tour dans un autre pays anglophone (USA), pour que l'interchangeabilité soit reconsidérée. Prenons l'exemple¹⁷ d'un indicateur lumineux, vert ou rouge, pouvant signifier « en surveillance » ou « en anomalie », suivant les pays. Un pays peut imposer d'harmoniser les couleurs entre tous les fabricants, par exemple le vert, et les modes d'emplois doivent alors être modifiés en conséquence. Pour les autres pays, la signalétique demeure indifférente. Mais ultérieurement, un de ces autres pays peut imposer le rouge. Il est évident que l'indifférence (l'interchangeabilité) antérieurement consentie n'est alors plus valable.

Nous allons essayer de généraliser les principes illustrés dans ces exemples. Tous les éléments constituent une population et il suffit d'établir des liens¹⁸ entre eux suivant une logique d'assemblage donnée (arborescence) pour constituer des sous-ensembles qui deviennent eux-mêmes des éléments, puis des ensembles de sous-ensembles (devenus éléments) qui correspondent aux objets techniques

14 Alphanumériques

15 Ensemble des dossiers de définition, données techniques, plans, modèles CAO, □

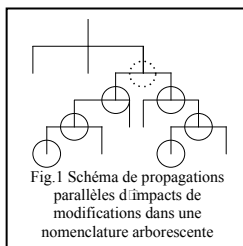
16 NATO (OTAN) & ISO Standards: □The ability of one product, process or service to be used in place of another to fulfil the same requirements□

17 Exemple emprunté à l'univers de la protection des biens et des personnes

18 Ces liens peuvent traduire une nomenclature d'assemblage, les fonctions que l'objet remplit au sein du système ou plus largement les moyens techniques conçus conjointement et/ou employés pour sa conception et sa réalisation.

voulus. L'identification d'un élément change¹⁹ avec sa définition²⁰. Aussi la population des éléments peut-elle s'enrichir de nouveaux arrivants nés de l'amélioration technologique de leurs aînés. Un sous-ensemble incorporant des modifications est une nouvelle sélection d'éléments modifiés. Un tel sous-ensemble devient lui-même un élément avec une identification qui le distingue de ses contemporains à moins qu'il ne soit considéré comme totalement interchangeable avec certains d'entre eux. Auquel cas il partagera la même identification. Dans ce formalisme, les exemplaires modifiés, qu'ils soient éléments ou ensemble d'éléments, sont distingués par une nouvelle identification, sauf s'ils sont considérés interchangeables, auquel cas l'identification sera identique et leurs compositions indifférentes. La conquête de l'interchangeabilité s'impose comme un moyen d'éviter la distinction systématique des générations successives d'éléments ou d'ensembles, qui segmente les séries et contrevient à la logique de sériation poursuivie. Mais c'est un moyen radical - sans mémoire de composition - par rapport au principe d'une gestion par rang ou par date consistant à faire l'appoint des modifications affectant un objet. La propagation des impacts d'un train de modifications dans les arborescences est-elle contrôlable avec la notion d'interchangeabilité?

La définition d'un élément, sous-ensemble ou ensemble, englobe en particulier, la liste des éléments qui le compose. Pour en savoir plus, il faut aller consulter la définition de chaque élément constitutif et ainsi de suite. Pour connaître l'antériorité d'un élément modifié, il faut cheminer en sens inverse. Retrouver dans sa composition, l'élément constitutif impacté qui a répercuté l'impact à son tour. Ainsi, la composition d'un élément, ou encore sa descendance, se découvre par ricochet. Une modification entraîne une nouvelle identification de l'élément et se répercute dans l'arborescence tant que l'interchangeabilité n'est pas retrouvée. Soit un nouvel objet est défini, soit l'interchangeabilité efface à un moment donné toute trace des différences de composition. La complexité de la propagation des impacts d'une modification par voisinage immédiat est amplifiée par le nombre d'éléments et de liens concernés ainsi que par le nombre et la fréquence des modifications. En pratique, l'étude séquentielle de chaque modification d'un train est impossible. L'exercice est de nature combinatoire et contraint à étudier plus de définitions qu'il n'en faut produire. L'étude simultanée d'un train de modifications est donc nécessaire, mais elle est incompatible avec une traçabilité des impacts. Impossible en effet de concéder une interchangeabilité pour une modification si elle est rompue par une autre du même



train. Et cela serait dépourvu d'intérêt. Impossible donc de tracer l'origine d'un changement d'identification quand plusieurs modifications l'ont provoqué. De plus, la progression de proche en proche masque les impacts transversaux (sur une autre branche de l'arborescence) et qui ne peuvent être évalués définitivement qu'au niveau des sous-ensembles communs. C'est à dire quand les processus d'analyse sont déjà très avancés impliquant de revenir sur d'éventuelles interchangeabilités concédées trop tôt.

En conclusion, ces modes de gestion des évolutions de définition très contraignants ne sont adaptés qu'à une cadence très modérée de modifications, c'est-à-dire pour une innovation contrôlée. Ici, il importe de bien faire remarquer que, dans une perspective de production en série, l'adoption d'un formalisme donnant aux éléments constitutifs d'un objet, leur autonomie en terme d'identification et d'évolution, implique soit des changements en cascade pour conserver la trace d'une composition modifiée, soit une interchangeabilité qu'on peut qualifier de totale puisqu'elle ne conserve aucune trace des modifications de sa composition dont on a considéré qu'elles ne rompaient pas l'interchangeabilité, deux alternatives incompatibles, comme nous venons de le voir, avec des vagues de modifications. Quand sériation nécessaire et innovation intensive se conjuguent, comment sont gérées les vagues de modifications dans le cas d'objets techniques complexes?

1.3. Relativiser les interchangeabilités en cas de recomposition combinatoire

19 L'identification étant commune à l'élément et à sa définition, toute modification se répercute physiquement par un marquage des objets, ainsi que sur les documentations associées et partagées par l'ensemble des acteurs.

20 Qui décrit sa composition en éléments et son mode de réalisation à partir de ces éléments (l'élément est l'article d'un ERP, d'une GPAO)

Les formalismes que nous avons décrits correspondent soit à des activités multi-unitaires (gestion des configurations et des évolutions par rang et date avec principe d'applicabilité), soit à des activités sériées où les rythmes d'innovation restent contrôlés (mise à jour des arborescences codifiées avec principe d'interchangeabilité). En fait, deux types d'activités ayant en commun une gestion des configurations qui ne soit pas de nature combinatoire. Soit parce que le nombre d'exemplaires est réduit et dans ce cas le faible régime de production peut s'accommoder d'un formalisme basé sur une définition monolithique des objets et un principe d'applicabilité, réclamé par un régime de conception accéléré. Soit parce le nombre de configurations est maîtrisé et dans ce cas le faible régime de conception peut s'accommoder d'un formalisme basé sur une définition des objets décomposable en éléments et un principe d'interchangeabilité, réclamé par un régime de production accéléré. Mais si dans ce dernier cas la complexité des objets techniques laisse tout au plus la possibilité d'étudier une seule configuration incorporant une vague de modifications, qu'il faut de plus réévaluer après chaque vague, le formalisme doit plier à ces contraintes, avec plus de souplesse que les deux précédents. Un tel formalisme est-il une combinaison des précédents ou de quelques autres transcriptions permises par les PLM pour ces usages hybrides? Est-il résolument nouveau et à concevoir?

La définition paramétrique de la validité commune [des modifications dépendantes en vue de la production] est fondamentale: elle permet de baser la validité d'un objet en fonction de la valeur de certains de ses attributs () la fonctionnalité de *mySAP PLM* constitue une mise en œuvre spécifique d'une stratégie de gestion des modifications. Nous sommes conscients que ce mode de fonctionnement ne convient pas à toutes les entreprises, toutefois il offre un point de départ satisfaisant pour la plupart des mises en œuvre. De nombreuses solutions²¹ () se targuent d'offrir des fonctionnalités d'analyse d'impact, permettant de retrouver les documents, données CAO, cahiers des charges, etc., susceptibles d'être revus suite à la modification d'un objet (d'un document, d'un article, etc.). Ces fonctions offrent des avantages considérables. Toutefois, de nombreuses solutions () ne parviennent pas à identifier les documents commerciaux ou de fabrication () habituellement gérés et conservés dans des systèmes ERP () En termes de gestion de la structure de produit, bon nombre de systèmes semblent offrir des fonctions élémentaires équivalentes. Il est toutefois important de contrôler si les fonctionnalités sont disponibles dans la solution de base livrée, ou si celle-ci tient plutôt de la boîte à outils, imposant au client de développer ses propres applications de gestion de structure de produit. (CIM Data, 2002). On voit bien qu'il n'existe aucune correspondance raisonnable entre les deux formalismes décrits jusqu'ici (*applicabilité* vs *interchangeabilité*). Les critères d'applicabilité et d'interchangeabilité n'ont rien à voir entre eux. Le premier est construit sur un rang ou une date variable mais avec un principe d'applicabilité exécutoire. Le second est construit sur une indifférence d'applicabilité à tel rang ou telle date, par principe même d'interchangeabilité, et par conséquent sur une indépendance avec les considérations de rang, date ou applicabilité. Une modification conduisant à une interchangeabilité totale n'a pas lieu d'être appliquée à un rang précis (sur le principe, cela équivaut à décréter une interchangeabilité valable pour tous les rangs). Si l'on souhaitait, pour gérer une activité hybride (sériée et fortement évolutive), adopter un formalisme lui-même hybride (applicabilité et interchangeabilité), cela impliquerait des choix d'options inconciliables que des circuits d'information parallèles viendraient nécessairement supporter ou supplanter. Dans un tel scénario en effet, la coexistence au sein des PLM, de toutes les options fonctionnelles nécessaires aux différents acteurs (conception, production) serait un transfert de pratiques, certes mutualisées par le collectif d'acteurs, mais néanmoins contradictoires. Avec à la clé, un travail amplifié par ce cumul de formalismes, nonobstant les risques de mésinterprétations et d'erreurs.

En septembre²² 2008, sur un A330, une configuration non conforme d'une des trois installations informatiques de contrôle de vol, a provoqué un atterrissage violent, au point de nécessiter le

21 Notamment citées dans le tableau (1) comparatif des stratégies de gestion des configurations multiformes,

22 Source Air Transport Intelligence news - 29/11/08

remplacement du train principal. Airbus et l'EASA²³ ont alerté les opérateurs de maintenance sur le respect des seules combinaisons autorisées par le constructeur: "To prevent an uncertified configuration that may result in unexpected operation of the aircraft systems owners and operators should adhere to the interchangeability and mixability rules given in Airbus type certificate holder documentation". En 2004, sur un Airbus A340, l'EASA avait signalé un problème identique de contrôle des manœuvres des aérofreins. Nous avons vu qu'une arborescence d'éléments peut se décrire par des éléments, des sous-ensembles et des ensembles (qui sont eux-mêmes des éléments)

Fig-2b - Schéma des configurations d'emplois d'éléments "multiformes"

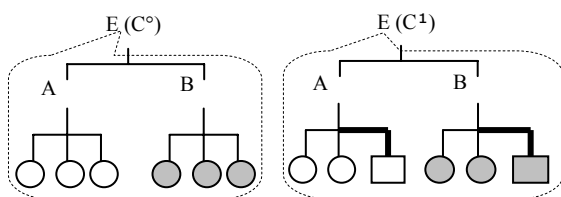
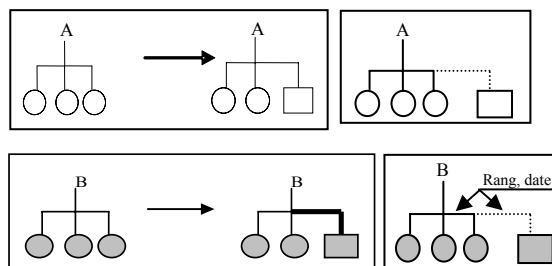


Fig-2a - Schéma de gestion des liens de nomenclature par rang ou date (sous-ensembles A et B)



relies entre eux d'une certaine façon²⁴. Lorsqu'un sous-ensemble A incorpore, sans changer d'identification, des éléments modifiés, il est indispensable de conserver la correspondance entre l'arborescence d'origine et la nouvelle (Cf fig-2_a et 2_b). Un lien existe donc entre le sous-ensemble et chaque élément d'origine. Un autre lien s'ajoute entre le sous-ensemble et chaque nouvel élément modifié. Gérer ces liens par rang et par date a

été expliqué. Il s'agit d'activer les premiers liens pour certains exemplaires puis de basculer sur les nouveaux liens pour les exemplaires suivants. Lors du basculement, les liens premiers sont désactivés. La gestion par date est identique, simplement le basculement est inscrit dans le temps. Ces basculements sont programmés pour écarter tout risque de mélange. Mais quand les deux liens coexistent, nous dirons qu'il s'agit d'un "multiformes". Si les deux formes sont totalement interchangeables, il est permis de les choisir indifféremment et la traçabilité des modifications est inutile. Imaginons un second sous-ensemble B vivant exactement le même scénario que le sous-ensemble A. Si un ensemble E coiffe les deux sous-ensembles A et B, il y a alors deux cas de figures:

- ❖ toutes les combinaisons entre les formes possibles de A et B sont étudiées pour garantir une totale interchangeabilité. Auquel cas l'ensemble E réunissant A et B conservera toujours la même identification E (et celle-ci sera suffisante). Mais nous avons vu que cette pratique était de nature combinatoire (avec le nombre de liens et d'éléments) et qu'elle est donc en général impraticable.
- ❖ une seule nouvelle combinaison est validée: A (avec le nouvel élément modifié) et B (avec le nouvel élément modifié). L'identification de E est inchangée (mais ne sera plus suffisante). L'interchangeabilité n'est valable qu'entre cette configuration validée et celle d'origine. Nous avons vu dans le cas de l'aéronautique, les exemples illustrant les répercussions induites d'emplois non-conformes (configurations non validées).

Un élément "multiformes" est donc un sous-ensemble ou un ensemble d'éléments, dont l'identification a été conservée alors même que sa composition a changé. Il peut regrouper différentes générations d'éléments dans des configurations autorisées bien déterminées. Pour gérer un élément "multiformes" il est nécessaire de gérer simultanément son identification avec ses configurations d'emplois (à savoir la configuration initiale C° de l'ensemble E et sa suivante C¹ du même ensemble E). Si l'identification de E avait changé, le problème ne se poserait pas et la gestion des cas d'emploi serait très simple, tout comme d'ailleurs la gestion des variantes et des options. Mais il serait trop coûteux de changer cette identification de E à chaque changement de composant (élément ou sous-ensemble). L'identification de E est donc inchangée et seule la connaissance des configurations C° et C¹ permet de distinguer les différentes formes de l'élément "multiformes". Les configurations étudiées et validées (C°, C¹) donnent les configurations d'emplois autorisées pour les

23 European Aviation Safety Agency

24 Cette modélisation est généralisable avec le langage de la théorie des graphes et plus largement encore dans la théorie des ensembles.

éléments interchangeables et garantissent également la coprésence des éléments modifiés. Au fil des évolutions de configuration, chaque élément est interchangeable relativement à une suite continue ou discontinue (C^0 , C^1 , C^i) de configurations successives autorisées, toutes interchangeables. C'est bien une interchangeabilité en valeur relative, *conditionnelle*, comparée à une interchangeabilité totale résultant d'une étude combinatoire. Mais évidemment, la seule considération de l'identification des éléments n'est plus suffisante pour assurer la présence simultanée des modifications d'une vague donnée, sans risque de panacher les éléments interchangeables. Il est indispensable de détenir la correspondance avec les configurations d'emplois autorisées (suite de configurations validées). Les *multiformes* sont par conséquent bien caractéristiques de formalismes hybrides qui ne peuvent être prohibés ou inhibés. *mySAP PLM* aide les utilisateurs à développer des structures d'articles () Ces structures peuvent être reliées entre elles pour définir complètement un produit donné. () Le concept de validité [date, numéro de série] permet de gérer la prise en compte des modifications de la structure du produit () [maîtrise de la configuration] il est possible de définir des configurations de produit et de gérer leur validité [le processus] doit être relié manuellement à chaque structure de sous-ensemble et à chaque document associé dans la structure du produit. () Très performantes, les fonctionnalités éprouvées de gestion de nomenclature d'articles de *mySAP PLM* peuvent toutefois s'avérer insuffisantes pour représenter des produits complexes présentant de nombreuses variantes. Dans certains cas, les relations avec les sous-ensembles utilisés deviennent confuses, ce qui rend l'utilisation du système malaisée. () (CIM Data, 2002). Les solutions PLM les plus élaborées s'efforcent en effet de contrôler les situations d'interchangeabilité conditionnelle²⁵ (*content interchangeability and use of context-driven content*). Elles permettent de définir des configurations de références (*baseline, photos, models*,), des *classes*, des *modules* ou des sous-ensembles (*subsets*) d'objets présentant une interchangeabilité totale [*"Full interchangeability", products that are identical in all of their technically relevant properties (Form, Fit, Function)*]²⁶ ou seulement restreinte [*"restricted interchangeability" mapped between Form Fit Function classes, with special group and conditions*]²⁷ that have to be met before products can substitute for one another]. Le degré variable de subdivision des objets permet de pouvoir aussi répartir la complexité structurelle sur différents niveaux (sous-ensembles, modules, méta-données, etc). Les stratégies se portent sur la gestion des liens (des relations) entre ces objets (resp. modules, etc). Mais elles se heurtent à la combinatoire des possibles et doit recourir à des règles de réduction du nombre de composants (Demoly, 2010; Boothroyd & Dewhurst, 1983). L'algèbre relationnel cherche alors à compenser l'inadaptation de la sémantique des identités et des dénominations. Il arrive que les identifications [*Manufacturer Part Number*] soient complétées de codes (*interchangeability codes; model identification*) communs ou distinctifs entre objets (resp. modules, etc) selon les configurations désirées. De tels codes sont généralement extensifs (dépendants du degré de subdivision des objets) et n'ont pas de repérages matériels. Le tableau ci-après présente une comparaison²⁸ des caractéristiques distinctives des formalismes proposés par les principaux²⁹ PLM du marché de l'industrie discrète (vs continue) pour gérer les objets *multiformes*. L'identité (via une interchangeabilité *totale* ou *restreinte*) et la différenciation (via les critères distinctifs d'identification³⁰ et/ou de liens) des formes d'objets générées sont variables selon notamment la subdivision considérée (en nomenclatures, en modules ou sous-ensembles nomenclaturés). La recherche des *identités* (resp. *différenciations*) à une échelle donnée (micro, macro, méta, etc.) déplace la recherche des *différenciations* (resp. *identités*) à un niveau inférieur ou supérieur. Ces formalismes complexes de gestion des configurations ne fonctionnent pas de manière entièrement

25 Soumise à conditions, ce qui a pour effet de restreindre son champ de validité (interchangeabilité restreinte vs interchangeabilité totale)

26 Physical, functional and performance characteristics or specifications that uniquely identify a component or device and determine its interchangeability in a system (Business Dictionary).

27 Par exemple, au voisinage des moteurs, seuls des produits hautement résistants à la chaleur peuvent être utilisés [Champion Aerospace LLC, Service Bulletin S.B.CH53536-1-74-001, Interchangeability and Intermixability of Parts, 19 décembre 2008].

28 A partir principalement de leurs descriptifs fonctionnels et/ou de retours d'expériences (club PLM Lab, CIM Data, notamment), ainsi que technologyevaluation.com (comparatifs sur 50 critères).

29 Notamment en termes de notoriété et/ou de parts de marchés dans l'industrie: Dassault Systèmes (25%), Siemens (19%), PTC (10%) selon le cabinet d'analyses américain Daratech en 2007,

30 Avec ou sans extension (codes)

automatisée et sollicitent visiblement une décision humaine (Brown, 2006; Bouikni et al. 2008; Hwang et al., 2009).

Tableau (1) comparatif des stratégies de gestion des configurations multi-formes (pour productions discrètes)							
Génération de formes				Modalités			
		<ul style="list-style-type: none"> ▪ Identité de formes ▪ Différenciation de formes ▪ Automatisation 				<ul style="list-style-type: none"> ▪ Type d'interchangeabilité [totale / restreinte] ▪ Critères distinctifs [identification / effectivité / modularité] ▪ Niveau [automatique / aide à la décision] 	
Solutions PLM	[identité des formes]	<input type="checkbox"/>	[Différenciation des formes]	<input type="checkbox"/>	[Automatisation]	<input type="checkbox"/>	<input type="checkbox"/>
✦ SAP (A&D)	☞ Lufthansa Technik.; NATO HelicopterInd.; Bombardier aéronautique; British Airways <input type="checkbox"/>						
	Interchangeabilité totale	<input checked="" type="checkbox"/>	identification	<input checked="" type="checkbox"/>	Effectivité ²	<input checked="" type="checkbox"/>	Automatique <input type="checkbox"/>
	Interchangeabilité restreinte ³	<input checked="" type="checkbox"/>	+ [code] ¹	<input checked="" type="checkbox"/>	Modularité ⁴	<input checked="" type="checkbox"/>	aide à la décision ⁵ <input checked="" type="checkbox"/>
✦ Lascom (ICS)	☞ Arianespace; OTAN; EADS; Thales <input type="checkbox"/>						
	Interchangeabilité totale ⁶	<input checked="" type="checkbox"/>	Identification ⁷	<input checked="" type="checkbox"/>	effectivité ⁸	<input checked="" type="checkbox"/>	Automatique <input type="checkbox"/>
	Interchangeabilité restreinte	<input type="checkbox"/>	+ [code]	<input type="checkbox"/>	Modularité	<input type="checkbox"/>	aide à la décision ⁹ <input checked="" type="checkbox"/>
✦ PTC (Windchill)	☞ GKN Aerospace Engineering Services; Oerlikon Solar; Schneider Electric; HP <input type="checkbox"/>						
	Interchangeabilité totale ¹⁰	<input checked="" type="checkbox"/>	identification ¹¹	<input checked="" type="checkbox"/>	effectivité ¹²	<input checked="" type="checkbox"/>	Automatique <input type="checkbox"/>
	Interchangeabilité restreinte	<input type="checkbox"/>	+ [code]	<input type="checkbox"/>	modularité ¹³	<input checked="" type="checkbox"/>	aide à la décision ¹⁴ <input checked="" type="checkbox"/>
✦ Dassault Sys. DS Portfolio (Transcat) ¹⁹	☞ Boeing; Airbus; Lockheed Martin Bombardier; Pratt & Whitney Canada; Dassault Aviation <input type="checkbox"/>						
	Interchangeabilité totale ¹⁵	<input checked="" type="checkbox"/>	Identification ¹⁶	<input checked="" type="checkbox"/>	Effectivité ¹⁷	<input checked="" type="checkbox"/>	Automatique <input type="checkbox"/>
	Interchangeabilité restreinte	<input type="checkbox"/>	+ [code]	<input type="checkbox"/>	Modularité	<input type="checkbox"/>	aide à la décision ¹⁸ <input checked="" type="checkbox"/>
✦ SIEMENS Portfolio ²⁰	☞ Lockheed Martin Aeronautics; B/E Aerospace; Pratt & Whitney; MBDA Missile Systems; GM <input type="checkbox"/>						
	interchangeabilité totale	<input checked="" type="checkbox"/>	Identification ²¹	<input checked="" type="checkbox"/>	effectivité ²²	<input checked="" type="checkbox"/>	Automatique <input type="checkbox"/>
	interchangeabilité restreinte ²³	<input checked="" type="checkbox"/>	+ [code]	<input checked="" type="checkbox"/>	modularité ²⁴	<input checked="" type="checkbox"/>	aide à la décision ²⁵ <input checked="" type="checkbox"/>
Extraits des descriptifs fonctionnels en annexe 1							

Jusqu'à quel point ces solutions sont-elles automatisables? L'obstacle le plus évident est la reconnaissance des différentes formes de l'objet physique dont l'identification est demeurée inchangée. Pour un opérateur donné, seule l'identification peut être parlante ou bien silencieuse sur la composition de l'objet. Or cette composition (sous-ensemble) n'est pas indifférente aux configurations de montage (sur-ensemble). Tant que cet obstacle n'est pas levé, toutes les potentialités des PLM (Merminod, 2007) ne peuvent être mises à profit. Il sera toujours impossible de calculer de manière automatisée le regroupement et la planification des besoins, ou de procéder à la recherche automatisée de l'ensemble des éléments sérialisés devant être remis à jour (retrofits), et de profiter pleinement des économies d'échelles. La survivance de différentes générations d'objets techniques se côtoyant dans les bases de données, les ateliers ou les parcs installés réclamera de les différencier (pour les éléments modifiés) et de les regrouper (pour les éléments invariants). Les PLM ne pourront que présenter des solutions incomplètes ou indécidables aux acteurs désignés, à des fins d'analyses et de décisions, avec un recours à une main d'œuvre coûteuse mobilisée à l'inspection des différences entre prévisionnel et opérations à réaliser réellement sur le terrain, entre éléments disponibles et identifications attendues, ainsi que leur localisation, ou encore au contrôle des écarts de configurations. Sans être nécessairement préjudiciables à la qualité des matériels, les inspections déboucheront dans les mêmes proportions sur des fiches d'anomalies, des dérogations et parfois même de délicates substitutions d'éléments. Il sera toujours aussi délicat de créer une campagne de mise à niveau d'un parc complet de matériels quelles que soient leurs versions respectives, de localiser les éléments ainsi que leur mouvement entre les installations et les magasins, ou de gérer la disponibilité des éléments. La vision globale et partagée des informations, permise par un outil homogène et communicant, constitue finalement une condition nécessaire (comment sinon gérer des relations entre objets évolutifs tout au long de leur cycle de vie?) mais pas suffisante. L'équivalence des relations entre sous-ensembles et sur-ensembles doit pouvoir se traduire à travers une

redéfinition de l'identité des objets (et par conséquent de l'identification). Cette question dépasse la population des PLM car elle nous renvoie notamment aux lois fondamentales du management des bases de données (Codd, 1970, 1990) qui mérite un détour.

1.4 Questionnements théoriques communs avec le management des bases de données: fondements de l'identité et de la discernabilité des ensembles

Tous les formalismes de gestion des configurations pouvant exister au sein des P.L.M et plus largement d'ailleurs au sein notamment des systèmes de gestion des bases de données³¹ ou même des outils de génie³² logiciel, s'évertuent à gérer dynamiquement les liens (relations, interfaces, frontières, etc.) entre les éléments (modules, sous-ensembles, etc.) résultant de la décomposition des objets (application, systèmes, etc.) et leurs générations successives (variantes, options, modèles, classes, etc.): *gestion de sources*³³, *outils de génération*³⁴, *intégration continue*³⁵, *environnement intégré*³⁶, etc. Les problématiques sont analogues et la littérature abondante (Raymond E.S., 1998; Bellagio et Milligan, 2005; Stark, 2004; Djeddar, 2003; Neagu et Faltings, 2001; Sacquet et Nowencien, 1995; Ghoul, 1983). Le National Institute of Standards and Technology (NIST) a estimé à 60 milliards de dollars par an les pertes enregistrées par l'industrie et le commerce américains à cause des bogues contenus dans les logiciels (S&T Press USA- n°324 - sept.2002). □The more complex the digital resource, the greater the potential loss is likely to be. For example, interchanging the data held in geographical information system (GIS) databases and groupware databases could involve the loss of thousands of links that have taken years of effort to create and which represent the bulk of the value of the database□(Feeney, 1999).

Dans un article fondateur sur les bases de données relationnelles de 1970, E.F. Codd aborde la problématique de la gestion des configurations de la manière suivante: □Many of the existing formatted data systems provide users with tree-structured files or slightly more general network models of the data. Application programs developed to work with these systems tend to be logically impaired if the trees or networks are changed in structure (□) Systems which provide users with a network model of the data run into similar difficulties (□) Activities of users at terminals and most application programs should remain unaffected when the internal representation of data is changed and even when some aspects of the external representation are changed.□ (Codd, 1970). L'interchangeabilité et les □multi-formes□sont au centre de cette problématique: □(□) The totality of data in a data bank may be viewed as a collection of time-varying relations (□). Accordingly we propose that users deal, not with relations which are domain-ordered, but with *relationships* which are their domain-unordered counterparts. In mathematical terms, a relationship is an equivalence class of those relations that are equivalent under permutation of domains□(Codd, 1970). Précisons qu'un domaine [d] est un ensemble dénombrable de valeurs caractérisé par un nom et qu'une relation [R] est un sous-ensemble du produit cartésien³⁷ d'une liste de domaines caractérisé par un nom. La concaténation (sans perte d'informations) de deux relations (R et S) ayant au moins un domaine commun, peut révéler un élément (□□) du domaine commun □(□) which gives rise to the plurality of joins. Such an element in the joining domain is called a *point of ambiguity* (□) A function is a binary relation, which is one-one or many-one, not one-many□(Tableau 2).

31 (BDD) informations stockées dans un dispositif informatique et organisées pour être consultées, modifiées, copiées, sauvegardées ou bien encore restaurées, suivant un modèle généralement relationnel. Un système de gestion de base de données (SGBD) est un ensemble de logiciels qui sert à ces opérations. Il est généralement utilisé simultanément par d'autres logiciels ainsi que les administrateurs ou les développeurs.

32 Ensemble des activités de conception et de mise en œuvre des produits et des procédures tendant à rationaliser la production du logiciel et son suivi [J.O; 19/02/1984]. Le site LWN.net a publié une analyse des contributions au noyau Linux sur un an (2.6.16 à 2.6.20): 28000 changements ajoutés par 1961 développeurs différents, remplaçant 1.260.000 lignes par 2.010.000 lignes de code nouveau, le noyau a augmenté de 754000 lignes.

33 Ajouts simultanés de centaines de développeurs, en identifiant précisément chaque modification, son auteur, sa date et sa finalité.

34 Pour automatiser les opérations de génération d'un programme, en gérant les dépendances entre composants.

35 Pour générer et contrôler l'ensemble de l'application, afin d'identifier le plus en amont possible d'éventuelles régressions, erreurs ou incompatibilités entre modules (résultant d'un découpage du système en sous-systèmes, classes, objets, fonctions).

36 Pour accueillir un grand nombre d'extensions

37 Un produit cartésien d'une liste de domaines [D1, D2, □, Dn] est l'ensemble des distributions de valeurs possibles (n-uplets) respectivement sélectionnées une à une dans chacun des domaines [(v1, v2, vn); (w1, w2, wn); etc.].

Domaine 1 (supplier)	(Common) Domaine 2 (part)	Domaine 3 (project)
Relation 1 (R)		Relation 2 (S)
1	1	2
2	1	1
2	2	1

question d'identifier de manière unique chaque ligne d'un tableau (éléments en relation), une « clé » [Primary Key] est nécessaire. Le plus souvent il s'agit d'une colonne additionnelle³⁸ attribuant des nombres identificateurs distinctifs, sans répétition ni case vide [subrogate primary key vs Foreign

Domaine 1 - $\varepsilon = \{A,B\}$	Domaine 2 - $\varepsilon^* = \{1,2,3\}$	$\varepsilon \times \varepsilon^*$	
A	1	A	1
B	2	A	2
Primary key	3	B	1
		B	2
		C	1
		C	2
		Foreign key	

La construction d'une relation se fait à partir d'un produit cartésien qui peut générer une relation de type «One-Many» comme le montre l'exemple ci-dessous (tableau 3). D'ailleurs, lorsqu'il est question d'identifier de manière unique chaque ligne d'un tableau (éléments en relation), une « clé » [Primary Key] est nécessaire. Le plus souvent il s'agit d'une colonne additionnelle³⁸ attribuant des nombres identificateurs distinctifs, sans répétition ni case vide [subrogate primary key vs Foreign key]. «Objects identified in one way: Both programming and non-programming users perceive all objects to be identified in exactly one way, whether these objects are abstract or concrete and whether they are so-called entities or relationships»(Codd, 1990).

L'unique et le multiple, la discernabilité, la dénomination et l'égalité (l'interchangeabilité) des choses dénommées sont des questions fondamentales tant pour l'algèbre d'E.F. Codd qu'en théorie des ensembles (Russel, 1903 ; Jech, 1978) sur laquelle cet algèbre s'appuie. «One important effect that the view adopted toward data has on the language used to retrieve it is in the naming of data elements and sets () The adoption of a relational model of data () permits the development of a universal data sublanguage based on an applied predicate calculus³⁹ () Predicate logic took 2,000 years to develop, beginning with the ancient Greeks who discovered that the subject of logic could be intelligently discussed separately from the subject to which it might be applied, a major step in applying levels of abstraction»(Codd, 1970). Il propose aussi une forme d'identification générique mais cette piste de réflexion demeure inachevée (nous la poursuivons en seconde partie) : «The simple form⁴⁰ $R.d$ will often be adequate () In the remainder of this paper, we shall not bother to distinguish between relations and relationships () while it is not the purpose of this paper»(Codd, 1970).

La théorie des ensembles définit la forme première de l'identité des objects: «Par un « ensemble » ou « système », j'entends en effet de façon générale toute multiplicité qui peut être pensée comme une unité, c'est-à-dire toute collection d'éléments déterminés qui peut être par une loi combinée en un tout» (Cantor, 1883). L'identité est conçue en passant de la «collection»⁴¹ à l'ensemble, du multiple [«many»; foreign key] à l'unique [«one»; primary key]. La conception de l'égalité (i.e. l'interchangeabilité) impose donc de repasser d'un univers à l'autre. Dans l'univers ensembliste en effet, deux choses identiques sont indiscernables: «Par *ensemble*, nous entendons toute collection M d'objets m de notre intuition ou de notre pensée, définis et distincts, ces objets étant appelés les *éléments* de M » (Cantor, 1895). «On peut expliquer un ensemble de manière imagée, en disant qu'un ensemble est une boîte « primaire » contenant des boîtes « secondaires » n'ayant jamais deux à deux de contenus identiques, ses éléments, contenant à leur tour des boîtes tertiaires contenant elles-mêmes, etc.» (Godement, 2001). La dénomination des choses égales renvoie au paradoxe de leur discernabilité: «Until predicates have been assigned, the two substances remain indiscernible; but they cannot have predicates by which they cease to be indiscernible, unless they are first distinguished as numerically different» (Russel, 1900). La conception d'une égalité est celle d'une nouvelle classe⁴² d'équivalence non dénommée a priori: «from similarity between elements it is

38 Identitycolumn pour SQL/Server, autonumber pour Microsoft Access, sequence pour Oracle, etc.

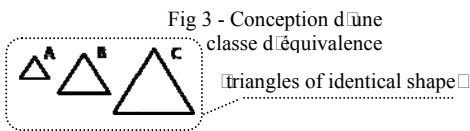
39 Définissant (via des symboles, variables, relations, connecteurs logiques, etc.) quels sont les énoncés valides et ceux qui ne le sont pas.

40 R est une relation et d un domaine

41 Le terme *collection* est généralement utilisé pour signifier un ensemble où l'ordre est ignoré, mais où la multiplicité est importante.

42 La notion de classe généralise celle d'ensemble. Dans le cas d'objets décrits sous forme de relations entre éléments (graphe), une classe d'équivalence est un ensemble.

possible to derive [by abstraction] another concept to which no name has yet been given. Instead of



the triangles are similar we say that the two triangles are of identical shape or the shape of one is identical with that of the other (Frege, 1893). Une telle conception par abstraction est située au point de passage entre la

collection (classes) et l'ensemble: The prevalent view is that abstracts should just be treated as equivalence classes (...) The theory of abstraction thereby becomes a part of the much more comprehensive theory of sets or classes (Fine, 2002).

En revenant aux origines de l'identité des objets, il est possible de reconsidérer les difficultés conceptuelles et pratiques de la gestion des configurations lorsque les régimes de reproduction et de transformation s'accroissent. Les stratégies de dénominations des objets et des classes d'équivalences peuvent offrir une alternative (ou une complémentarité) aux stratégies de gestion des relations.

2. La conception d'une identité relative et les stratégies gestionnaires

La gestion des relations s'accommode mal de la question de l'interchangeabilité de deux ensembles (représentant des objets, etc.), autrement dit de leur identité, car elle traduit une quête d'indépendance à l'égard de leurs contextes respectifs. D'ailleurs, l'égalité de deux ensembles (autrement dit leur identité) s'établit par inclusion mutuelle, chaque ensemble servant de contexte à l'autre : si A est inclus dans B et que B est inclus dans A alors A est égal à B et réciproquement. La quête d'indépendance suppose ainsi, par définition, de pouvoir s'abstraire de toute relation de dépendance contextuelle. Il serait donc paradoxal de rechercher parmi les stratégies de gestion des relations seulement, un moyen d'automatiser le traitement des interchangeabilités et donc des configurations. Nous proposons par conséquent d'explorer des stratégies alternatives de gestion des dénominations et de formuler les conditions que nous pensons nécessaires et suffisantes pour qu'un traitement automatisable au sein des P.L.M. devienne possible. C'est d'ailleurs bien une des aspirations des P.L.M. que de proposer des mécanismes notamment d'identification et de modélisation des informations associées aux produits et aux process (Abramovici, 2007; Reix et Al, 2011). Nous proposons de revenir d'abord sur la question de l'identité et de la discernabilité, en distinguant ces deux concepts fusionnés en théorie des ensembles. Nous abordons ensuite la question de l'interchangeabilité dans sa capacité à absorber les évolutions de configurations en reconcevant des invariances conditionnées dans leur identité et leur discernabilité. Cette question est notamment illustrée par des millésimes historiques que sont l'avènement de l'imprimerie (standardisation par la conception d'une manière de mouler pour reproduire des caractères multiples et interchangeables) et de l'horlogerie (conception de la mère des machines pour absorber les effets d'une innovation). Pour finir, nous présentons les principes d'une typologie des correspondances entre certaines dynamiques industrielles (repérables d'après les régimes de reconception et de reproduction) et les stratégies susceptibles de leur convenir pour gérer les évolutions de configurations (gestion des relations, gestion des identités).

2.1 Identité et discernabilité des ensembles, une conception de l'invariance

Les stratégies de gestion des évolutions de configurations portent sur les relations entre éléments, sous-ensembles et ensembles⁴³ et sur les dénominations qui leur sont attribuées. Les lois d'identification et de composition⁴⁴ des entités devant constituer les objets désirés découlent des stratégies de choix⁴⁵ opérables sur des populations de liens et de dénominations se renouvelant à un rythme plus ou moins important. Dans l'univers de la chimie, de la biologie ou de la génétique, ces

43 Reflétant par exemple le découpage des objets en composantes pouvant évoluer les unes par rapport aux autres (nomenclatures, interfaces, etc.).

44 Loi de composition interne mathématiquement parlant [(re)composition de couples de sous-ensembles dans un ensemble non vide].

45 Un choix peut se traduire avec une fonction définie sur un ensemble d'ensembles et qui à chacun d'entre eux associe un de ses éléments.

lois peuvent différer substantiellement de l'univers du logiciel ou de l'aéronautique, les normes, les interprétations et les usages étant relativement différents. Il s'agit néanmoins de trouver un algèbre et une sémantique permettant aux concepteurs⁴⁶ de décrire les états successifs de la définition d'objets, dont les générations se côtoient, avec leurs changements et leurs invariances (Mantripragada et Whitney, 1999). L'échelle de temps interagit évidemment sur la capacité à gérer des représentations structurellement adaptées, identifiables, reconnaissables et reproductibles pour les différents acteurs. La "philosophie de la nature" est, selon Aristote⁴⁷, "l'étude des choses qui changent" (Kosman, 1969).

La question première est celle de l'identité d'un objet et de l'égalité de deux objets (objets identiques, de même identité): "qualité d'être bien celui qu'il prétend être" et "qu'une chose est la même qu'une autre" (Littre, 1976). Le concept même d'ensemble (au sens de la théorie des ensembles) se conçoit dans l'unité à partir d'une *collection* originelle où les identités peuvent être multiples. "Je nomme de telles multiplicités, multiplicités absolument infinies ou inconsistantes [impossible à concevoir comme une unité, comme un objet achevé] (□) Si, au contraire, la totalité des éléments d'une multiplicité peut être pensée comme existant simultanément, de telle sorte qu'il soit possible de la concevoir comme un seul objet, je la nomme une multiplicité consistante ou un ensemble⁴⁸" (Levy, 1987). Le concept de "multiformes" n'appartient pas à l'univers des ensembles mais à celui des collections d'où l'absence d'interchangeabilité totale entre les configurations "multi-formes" "J'appelle ensemble, une collection à laquelle nous imputons un concept tel que l'arrangement des parties⁴⁹ soit indifférent (dans lequel rien d'essentiel n'est donc changé pour nous lorsque seul l'arrangement est modifié) ; et j'appelle pluralité A un ensemble dont toutes les parties sont considérées comme des unités d'une certaine espèce A, i.e. comme objets subsumables sous le concept A" (Bolzano, 1993). En générant les "multiformes", les concepteurs ouvrent ainsi à la frontière entre *collections* et *ensembles*. Ces concepts paradoxaux du point de vue des ensembles, puisqu'il s'agit de configurations jugées équivalentes mais pas indépendamment des contextes d'emplois respectifs, émergent quand le rythme des évolutions s'accroît et que le renouvellement des dénominations est impraticable (Cf 1.2). Si l'interchangeabilité est relative, contextuelle et donc conditionnelle, des dénominations figées deviennent silencieuses sur les relations autorisées entre sous-ensembles et ensembles et le choix (notamment physique des objets) est ingérable.

L'équivalence de formes, non discernables par leurs dénominations ("multiformes"), relativise l'égalité classique dite "des indiscernables" au sens de Leibnitz : "(□) sont les mêmes choses dont l'une peut-être substituée à l'autre sans dommage pour la vérité" (Leibnitz, 1714, 1998). Cette égalité est absolue. Deux ensembles distincts ne peuvent avoir les mêmes éléments et deux ensembles égaux ont les mêmes propriétés. La stratégie de dénomination est alors purement arbitraire: "Les propositions de la forme $a=b$ ont bien souvent un contenu fort précieux pour le progrès de la connaissance, et elles n'ont pas toujours un fondement *a priori*. La découverte que chaque matin se lève le même soleil, et non pas un nouveau soleil, a bien été une des découvertes les plus fécondes de l'astronomie (□) J'emploie ce mot [dénotation] au sens d'identité et je comprends « $a=b$ » au sens de « a est le même que b » ou « a et b coïncident » (□) la proposition $a=b$ ne concernerait plus la chose même, mais la manière dont nous la désignons." (Frege, 1879; 1994).

Mais une telle relation d'égalité n'est pas toujours constructible pour des concepteurs, y compris sur le plan mathématique (Bridges et Reeves, 1999). En effet, "(□) dire que $a=b$ si et seulement si a et b se comportent de même dans "tout contexte"(□) est en fait équivalent au cas général [l'égalité de Leibnitz]" (Girard, 2009). Ainsi, certaines questions sont indécidables comme par exemple le

46 On englobe sous cette désignation l'architecture système (sous-systèmes, interfaces) [Le Moigne, 1977],

47 ~ -350 a.c.

48 Propos cités de Georg Cantor

49 Au sens de sous-ensemble

problème de l'équivalence des programmes (deux programmes informatiques étant donnés, calculent-ils la même chose ?) ou le problème de l'utilité d'une partie d'un programme (l'ensemble de codes constituant un programme informatique donné, contient-il un sous-ensemble de codes inutile, c'est-à-dire qui ne sert jamais, quelle que soit l'utilisation faite du programme)? Il n'existe pas d'algorithmes (Turing, 1936, 1948; Rice, 1953) permettant de trancher par oui ou par non en un nombre fini d'étapes (il est impossible de trouver une méthode traitant tous les cas systématiquement).

Le "multiformes" résulte de la conception de classes d'équivalences constructibles mais fondées sur une relation d'égalité relative plutôt qu'absolue, donc plus restrictive⁵⁰ sur les contextes possibles (configurations favorables parmi la combinatoire des agencements possibles). Un objet est reconnaissable par rapport à un autre, avec lequel il est ou non interchangeable, relativement aux contextes connus pour autoriser une telle substitution. Il n'est plus question que de classes d'équivalence d'objets relativement à des classes d'équivalence de contextes. La nature "relative" de la discernabilité suppose une stratégie de dénomination plus générale que la dénomination arbitraire classique pour représenter les classes d'équivalences possibles: "[a=b] On ne saurait les distinguer que si la différence des signes [a ; b] correspond à une différence dans la manière dont l'objet désigné est donné. (□) Par suite, la proposition contient une connaissance effective." (Frege, 1884).

Une classe des compositions interchangeables (c_i) et une classe de configurations d'emplois interchangeables (C_i) permettent de construire une stratégie de dénomination de la forme $c_i^* | C_i^*$ avec c_i^* et C_i^* représentant les classes d'équivalences correspondant aux c_i et aux C_i respectivement. Littéralement, $c_i^* | C_i^*$ se lit : "est une composition quelconque de la classe d'équivalence représentée par c_i^* sachant⁵¹ son emploi possible dans une configuration quelconque de la classe d'équivalence représentée par C_i^* ". Cette forme $c_i^* | C_i^*$ est nécessaire et suffisante en informations pour identifier chaque élément et reconstituer les sous-ensembles et les ensembles désirés (Fig.4). Les concepteurs des entreprises étudiées⁵² procédaient ainsi, repérant les éléments invariants et ceux modifiés entre deux états de la configuration (initial et final), puis en établissant des modèles⁵³ de correspondances (Cf 1.3) entre les compositions interchangeables (c_i) et les configurations d'emplois interchangeables (C_i). Ce formalisme traduisait de manière entièrement automatisable l'algèbre relationnel que les concepteurs adoptaient pour absorber les vagues de modifications impactant les objets. Mais les règles normées d'identifications et d'évolutions étant généralement construites sur des relations d'appartenance⁵⁴ (à des familles, classes ou catégories de produits) plutôt que sur des classes d'équivalences, une double correspondance était néanmoins nécessaire.

Fig.4 □Stratégies de dénomination

- Schéma standard: Etat initial (ensembles structurés d'objets dénommés) → Dénominations inchangées (interchangeabilité absolue) ou nouvelles dénominations normées (rupture d'interchangeabilité) → Etat final
- Nouveau schéma générique: Etat initial (ensembles structurés d'objets dénommés) → Conception de classes d'équivalence relatives → Dénominations génériques $c^* | C^*$ [ou $d^*.R^*$] → Etat final

La forme générique $c_i^* | C_i^*$ fait écho à celle pensée par E.F. Codd *R.d* (Cf.1.4) à condition d'interpréter R comme un représentant de la classe des relations (de composition, etc.)⁵⁵ équivalentes malgré les permutations de domaines d (d s'interprétant alors comme un représentant de la classe d'équivalence des domaines permutable). Cette forme générique est compatible avec les

50 Polymorphisme selon Girard P.Y. (2009)

51 Dans le même sens que la conditionnalité employée par Bayes en statistiques [Bernardo J.M. & Smith A.F.M., (1994), *Bayesian Theory*. Wiley,]. La dénomination standard suppose l'interchangeabilité des configurations d'emplois absolue d'où la stratégie de dénomination classique: $c_i^* |$ sans conditions, avec $c_i^* = c$ (pas de distinction de dénomination entre la collection d'objets absolument interchangeables et l'exemplaire lui-même)

52 Notamment au sein du Système d'Information Technique Industriel dédié à l'activité spatiale et de l'armement de l'entreprise X.

53 Dénommées "programme modèle" (dictionnaire interne de l'activité "espace & défense" Giacomoni G., 1993).

54 American Standard Code for Information Interchange (ASCII); (NATO) International Standard Organization (ISO); Unified Modeling Language (UML) 2.0; System Modeling Language, etc.,

55 Une composition peut se traduire pas une relation entre composé et composant.

outils P.L.M. ou du génie logiciel et des SGBD. Du reste, elle traduit l'identité des objets comme une régénération permanente de classes d'équivalence, représentées par "abstraction" au sens entendu par Frege (1893) et Fine (2002) notamment (Cf.1.4).

2.2 L'interchangeabilité, une capacité à préserver les invariances ou à en concevoir de nouvelles

Si l'on se réfère au Larousse, la première occurrence de l'interchangeabilité apparaît en 1931 pour caractériser les pièces standardisées, fabriquées en séries. L'adjectif interchangeable lui est antérieur, 1870 en France, 1450 en Angleterre. Il est employé pour les pièces et objets semblables, de même destination, et qui peuvent être changés l'un pour l'autre ou mis à la place les uns des autres (pneus interchangeables, mécanismes à pièces interchangeables, etc.). L'interchangeabilité a pu être considérée comme *totale* ou *restreinte (restricted interchangeability)*. La restriction laissant entendre, que dans certaines limites de validité connues, l'interchangeabilité peut être parfaitement et définitivement établie. Sans nécessairement toujours épouser exactement la même teneur, l'interchangeabilité restreinte semble avoir émergé dans différents domaines, en sciences du vivant (Micklem et al., 1975), en statistiques sous forme de tests⁵⁶ de proportions de valeurs (et modèles) interchangeables (Jerdack et Pranab, 1990), en mathématiques appliquées à l'industrie (Steiner et Stephenson, 1990), en informatique industrielle (Dirickersbach, 2008). L'interchangeabilité en valeur *relative* et *conditionnelle* (Wiggins, 2001; Geach, 1980; Giacomoni, 2002) et non plus en valeur absolue, est une reconception profonde de l'identité et de la mutation des objets techniques. Considérer que toute *interchangeabilité* est relative à un état de l'art et des connaissances (Simon, 1976) et ne peut être intemporelle, revient à dissocier le concept d'*interchangeabilité* de la seule propriété qui lui ait été historiquement octroyée: "être absolue". Il en est ainsi de la relation d'égalité, toujours considérée comme définitivement acquise, fût-elle totale ou bien restreinte, alors même qu'elle n'est pas toujours constructible pour "tout contexte".

On peut distinguer deux espèces d'interchangeabilité: L'élément est invariant dans un ensemble différent (première espèce) [Ce qui correspond aussi bien que au cas d'un élément commun à plusieurs objets sans aucun liens de parenté, qu'au cas d'un élément commun à plusieurs générations d'un même objet à l'origine]; L'élément change et l'ensemble est invariant (deuxième espèce). Ces deux espèces d'interchangeabilité ont chacune leur traduction en termes de cas d'emploi:

- emploi d'éléments identiques sur des ensembles (objets) de conception ou de fabrication différente, pour l'interchangeabilité de première espèce.
- emploi d'éléments de conception ou de fabrication différente, sur des ensembles (objets) identiques, pour l'interchangeabilité de seconde espèce.

Elles ont aussi chacune leur traduction industrielle et leur histoire. Première moitié du XV^e siècle, l'imprimerie associe la presse à imprimer⁵⁷, mise au point d'après le modèle des pressoirs à vis, et les caractères métalliques mobiles⁵⁸. L'imprimerie (Chapell, 1970; McMurtie, 1942) est essentiellement un perfectionnement inspiré d'autres⁵⁹ inventions de l'époque notamment dans le domaine de la métallurgie (fabrication des poinçons; jets en moules). L'avancée décisive de Gutenberg⁶⁰ (Updike, 1920; Scholderer, 1970; Lehmann-Haupt, 1966) est la lettre métallique moulée et réutilisable (les *types*) qui seule répond à l'exigence d'une impression sans "foulage". Il faut, pour compenser les irrégularités du moulage (Bertrand, 1787), exercer une forte pression qui marque le verso du papier et en interdit l'impression. Les *types* solutionnent une problématique d'interchangeabilité de seconde espèce (Fig.5 annexe 2): la standardisation des éditions⁶¹

56 Non paramétrique

57 Qui existait déjà avec « le presse » pour la gravure sur bois (xylogravure)

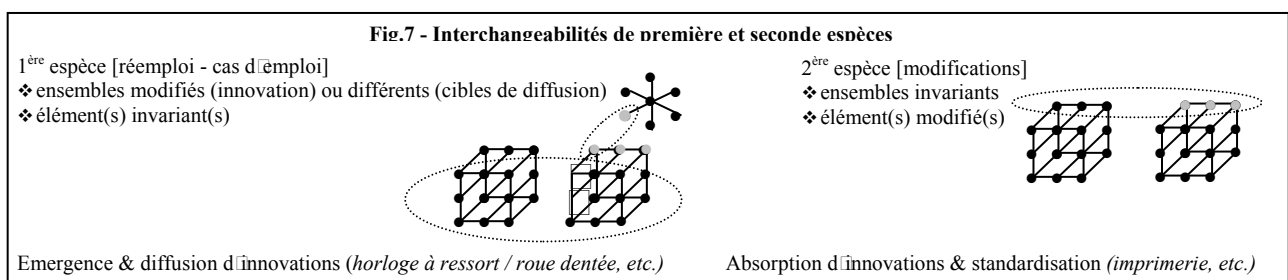
58 Qui existaient au moins depuis le XI^e siècle (vraisemblablement dès le II^e) en plomb, puis en cuivre - Bi Sheng (1041-1048).

59 Dont le papier (qui apparaît en Chine au début du II^e siècle) et l'encre permettant l'impression des deux faces du papier.

60 Orfèvre de formation.

61 La première est le Psautier de Mayence (1457). Les éditions sont du reste hybrides, imprimées et manuscrites.

imprimées. Fin XV^e et XVI^e siècle, l'horlogerie (Howse, 1980) adopte un nouveau système moteur, la force élastique du ressort, en remplacement d'un poids suspendu à une poulie. L'instrument horaire peut ainsi devenir portatif. Cette innovation soulève la question du réemploi des autres éléments demeurés invariants (interchangeabilité de première espèce □ Fig.6 annexe 2). Pour compenser les variations de force du ressort, l'horloge utilise un cône cannelé spécial nommé *fusée* dont le pas de vis est difficile à faire à la main. La taille à la main ne permet ni espacements précis ni découpages nets. J. Ramsden⁶² (1735-1800) élabore un tour permettant d'obtenir la vis-mère (vis de contrôle) nécessaire à sa *machine à diviser*⁶³ (Ramsden, 1787) de haute précision, réutilisée pour nombre d'instruments scientifiques: sextant, baromètre, microscope, télescope, etc. L'horloge est le premier appareil de mesure moderne et les horlogers sont les premiers fabricants d'instruments scientifiques. Ils fondent⁶⁴ la technologie des machines outils, en particulier dans trois domaines: la roue dentée, le ressort et la vis. La question des cas d'emploi de ces éléments sur des produits différents est celle de l'interchangeabilité de première espèce. L'horlogerie devient ainsi "mère des machines" décloisonnant les divers champs de connaissance et de savoir-faire. L'imprimerie et l'horlogerie ont en commun le passage de productions manuelles (manuscrit; taille manuelle des rouages) à des productions (partiellement) automatisées (caractères mobiles moulés; machine à diviser). Les avancées technologiques (vis, ressort, roue dentée) issues de l'horlogerie se diffusent aux instruments scientifiques. L'imprimerie accueille et perfectionne des avancées technologiques (poinçon; moule) issues de la métallurgie. En corollaire, l'interchangeabilité traduit la capacité de préserver les invariances ou d'en concevoir de nouvelles: en émergence et diffusion (1^{ère} espèce) / en absorption⁶⁵ (Saga et Zmud, 1996) et standardisation⁶⁶ (2^{nde} espèce). Historiquement, de nombreuses mutations industrielles sont marquées par l'absorption d'innovations technologiques et la standardisation des techniques et des méthodes (Cohen, 1994). Somme toute, la sériation exige une stabilisation technologique que remet en cause l'innovation. Le schéma ci-après (Fig.7) décrit les processus d'évolution des objets suivant que l'on considère les éléments impactés (nous parlons alors d'interchangeabilité de seconde espèce) ou les éléments invariants (nous parlons alors d'interchangeabilité de première espèce). Qu'il s'agisse des caractères mobiles et interchangeables pour imprimer les pages d'un livre ou des roues dentées interchangeables entre une horloge et un microscope, la rupture d'interchangeabilité marque la fin d'une reproduction⁶⁷ d'objets identiques ou transposables (Huang, 1996). Avec pour conséquences une diversification, une segmentation des séries et une crise potentielle du formalisme gestionnaire (Hatchuel et Sardas, 1992). Il nous faut constater que les éléments rencontrent des contextes différents lorsqu'ils sont employés sur des objets qui évoluent, que les objets rencontrent des contextes différents lorsqu'une innovation se diffuse plus largement, que les contextes et les contenus sont relatifs et varient tantôt séparément tantôt simultanément.



62 Mécanicien de précision inspiré de la technique des premiers horlogers français (Lenoir; machine à fusée; 1741).

63 Machine à diviser les instruments de mathématiques

64 La première machine à découper les rouages est l'œuvre de J. Torriano (1540) pour construire une grande horloge astronomique (20 ans de mise au point, 1800 rouages, et 3,5 ans de réalisation).

65 Absorption par un milieu d'une innovation de source externe (exogène). La capacité du milieu à absorber une technologie et à développer des usages innovants étant essentiellement fonction de la connaissance du métier (et donc des cas d'emplois) plus que de la technologie elle-même, ce que montre bien R. Zmud. Le schéma proposé autorise une interprétation duale en termes de connaissances.

66 NATO (OTAN) & ISO Standards: "The development and implementation of concepts, doctrines, procedures and designs in order to achieve and maintain the compatibility, interchangeability or commonality which are necessary to attain the required level of interoperability, or to optimize the use of resources, in the fields of operations, materiel and administration"

67 Règle Design for Assembly n°3: Standardiser les composants [réduire le type de composant]

Comment maîtriser l'identité (l'égalité) des objets pour prévenir des ruptures d'interchangeabilité, qui peuvent transformer le visage de l'activité et rendre inapte une stratégie de gestion des évolutions de configurations ? Comment à dessein, concevoir les objets en anticipant leur renouvellement et mieux absorber les effets de l'innovation ?

2.3 Stratégies de gestion des évolutions de configurations optimisées en fonction des dynamiques de reconception et de reproduction

Nous avons vu que l'interchangeabilité est une invariance de contextes (configurations) ou bien de contenus (Cf 2.2). C'est une identité relative et conditionnelle (Cf 2.1). Nous avons jusqu'ici raisonné sur des objets (des ensembles) dont la décomposition⁶⁸ en éléments et sous-ensembles était supposée déterminée. Mais il nous faut aussi raisonner à décomposition variable sous l'effet des régimes de reconception et de reproduction. Une définition d'ensemble de l'objet, sans définition autonome de chacun de ses éléments, peut convenir à condition que cette définition soit figée. En revanche, comme nous l'avons vu, si la définition change, parce qu'un élément au moins évolue, un problème va se poser pour reproduire l'objet en série, parce que les éléments invariants et les éléments modifiés sont totalement solidarisés dans une définition monolithique. Il sera opportun de décomposer la définition d'ensemble pour que chaque élément constitutif ait sa propre définition, de sorte qu'il puisse évoluer de manière autonome. Il est nécessaire que les éléments variables puissent être séparés et distingués des éléments invariants. La séparabilité d'un élément consiste à lui donner une existence autonome (identité propre, définition et reproduction autonomes).

✦ *Principe [premier] de préservation des invariances par séparabilité des éléments variables (ou principe du départitionnement)*

Le principe du partitionnement se manifeste par exemple dans la conception des systèmes logiciel/matériel, par la séparation «matériel⁶⁹ (invariant) / logiciel⁷⁰ (variable)» avec une décomposition multicouches. Ou encore, dans la séparation entre la conception des couches de communication et les parties matérielles et logicielles (Rousseau, 2005). Ce principe peut permettre de reproduire les éléments invariants en réalisant des économies d'échelle (et donc de fabriquer en série) ou encore de limiter la segmentation des séries dans une économie de variété. Les relations au sein d'un système et entre le système et son environnement peuvent être sensibles aux modifications affectant tout ou partie du système ou de son environnement. Les relations d'équivalence (d'invariance ou d'interchangeabilité) permettent de partitionner un système en regroupant les entités constituantes qui sont en relation afin de constituer des classes d'équivalence étanches (deux classes ayant une entité commune doivent fusionner). Le principe du départitionnement veut prévenir toute potentielle rupture des relations d'équivalence connue entre une entité et les autres lorsque son état est destiné à être partiellement modifié. La partie modifiée est alors séparée de la partie invariante. Mais cette appréciation de l'invariance et de ses frontières est relative à la fois à la nature des modifications devant survenir et à la nature de la relation d'équivalence déjà existante avec les autres entités. La connaissance du système et de son environnement conditionne donc l'application de ce principe qui conduit du reste à subdiviser de manière expansive, aussi loin et aussi souvent que les régimes de reconception et de reproduction le nécessitent. La question de l'invariance (ou de l'interchangeabilité) peut se poser de deux façons:

❖ comment reconnaître (et identifier en conséquence) les éléments (ou sous-ensembles) demeurant interchangeables malgré les évolutions internes (seconde espèce), c'est à dire laissant invariants

68 Product Breakdown Structure

69 Processeurs, composants spécifiques, mémoire, réseau de communication complexe, □

70 Pilotes, interruptions, gestion des ressources, interfaces logiciel/matériel, logiciels de l'application, □

tous les autres éléments de leur environnement lorsqu'ils sont combinés⁷¹ avec eux (relation fonctionnelle, etc.) ?

- ❖ comment reconnaître (et identifier en conséquence) les éléments (ou sous-ensembles) demeurant interchangeables malgré les évolutions de leur environnement (première espèce), c'est-à-dire ceux que tous les autres éléments de leur environnement laissent invariants lorsqu'ils sont combinés⁷² avec eux (relation fonctionnelle, etc.) ?

Cela conduit au second principe de constructibilité de nouvelles invariances avec un langage et un algèbre adéquats:

✦ *Principe [second] de constructibilité des invariances par conception d'un langage des dénominations extensible aux identités relatives (ou principe du repartitionnement)*

Ce principe explique que pour pouvoir concevoir des classes d'équivalence nouvelles entre éléments ou sous-ensembles, soit entre les générations successives d'un même objet, soit entre des objets n'ayant à l'origine aucun lien de parenté mais ayant incorporé au cours de leur cycle de vie, des éléments (ou sous-ensembles) identiques (souvent après diffusion d'une innovation), il faut redéfinir les conditions d'interchangeabilité afin de pouvoir reconnaître et identifier les éléments (ou sous-ensembles) en conséquence. Les stratégies de gestion des dénominations et des relations sont conçues et appliquées à dessein pour gérer les évolutions de configurations. C'est donc sous la forme d'un langage de dénomination générique que l'identité des objets a été formulée (Cf 2.1). De sorte qu'elle ne soit plus définie indépendamment de la vie de l'objet comme c'est généralement le cas, mais au contraire d'après l'historique des compositions et des configurations d'emplois, lesquelles résultent respectivement des évolutions internes et environnementales. L'histoire identitaire d'un objet permet de mieux concevoir ses futurs possibles. Comme le disent Jean-Pierre Changeux et Alain Connes "puisque un langage est fait pour reproduire" (□) "Il a aussi un caractère prédictif" (Changeux et Connes, 1989).

La constructibilité des invariances se pose avec les changements de contextes qui font apparaître des variabilités, mais aussi avec l'élargissement ou plus exactement l'extension de contexte. Le terme d'extension prend ici le même sens qu'en mathématiques lorsqu'il est question de plonger un espace d'objets (ou de données) dans un espace plus général. Les nouvelles conditions d'interchangeabilité ne font pas table rase des connaissances préexistantes ayant permis de définir les conditions d'interchangeabilité initiales. Par exemple, l'espace \mathbb{N} des nombres entiers est plongé dans l'espace \mathbb{Q} plus vaste des nombres rationnels. La transition (Cf 1.4) s'opère à partir des collections du produit cartésien $\mathbb{N} \times \mathbb{N}$ et dans \mathbb{Q} tous les nombres sont redéfinis comme des couples d'entiers (a,b) en relation fractionnaire $(\frac{a}{b})$ avec un dénominateur b non nul. Les nombres entiers sont reconstruits sur ce nouveau schéma comme des couples $(a,1)$ et ont alors deux dénominations coexistantes⁷³ pour désigner la même entité, a [relativement à \mathbb{N}] et $(a,1)$ relativement à \mathbb{Q} . L'identité (l'interchangeabilité) des nombres est redéfinie dans \mathbb{Q} : deux rationnels, à savoir deux couples d'entiers (a,b) et (c,d) avec b et d non nuls, sont égaux si et seulement si $a.d = b.c$. En effet, $(2,1)$ est identique à $(4,2)$ ou à $(8,4)$, etc. Dans \mathbb{Q} , la dénomination $(2,1)$ représente la classe d'équivalence de tous les couples ainsi générés. L'extension est formulée sous le terme d'abstraction par Frege (1893) et Fine (2002) lorsqu'il est question de concevoir de nouvelles classes d'équivalence non encore dénommées (Cf 1.4). L'extension concerne tous les

71 Principe d'élément neutre d'une loi de composition interne

72 Principe d'ensemble stabilisateur

73 La première étant présente dans la formulation de la seconde

objets⁷⁴ autres que des nombres (Hatchuel et Weil, 2007) et l'abstraction est notamment pratiquée dans le développement logiciel (Brunet, 1991; Lee et Al., 1992; Brönnimann et Al., 2009).

La conception des invariances, tout comme l'ajustement permanent des stratégies de gestion des évolutions de configurations aux dynamiques combinées de reconception⁷⁵ (innovation) et de reproduction⁷⁶ (sériation), reposent évidemment sur des collectifs humains (et donc l'organisation) intervenant tout au long des cycles de vie des systèmes (Carbonel, 2001; Patout, 2001; Hussenot, 2007). De nombreux travaux (Chandler, 1962, 1977; Woodward, 1965; Perrow, 1967; Lawrence et Lorsch, 1967; Giard, 2003; Bourdon et Lehmann-Ortega, 2007) ont du reste mis à jour le rapport des systèmes et des organisations avec les évolutions de l'environnement. Les deux principes précédemment énoncés peuvent contribuer à une meilleure compréhension des logiques d'ajustement des stratégies de gestion des évolutions de configuration aux spécificités de l'activité.

La complexité de la gestion des évolutions de configurations concerne les ensembles et dépend essentiellement de trois facteurs interagissant : la cadence de reproduction (en série), la cadence de reconception (renouvellement des définitions) et les cycles de vie⁷⁷. En corollaire, la mémorisation⁷⁸ des configurations nécessaires et suffisantes (destinées à être reproduites ou maintenues), sur un horizon qui dépend pour sa part des cycles de vie, nécessite d'être optimisée par des stratégies de gestion des dénominations et des relations adaptées. On peut comprendre facilement par exemple, que des objets fabriqués en petites séries et fréquemment renouvelés, puissent vivre sous un nombre grandissant⁷⁹ de définitions différentes. Une population de tels objets inter-reliés par une structure d'assemblage ou de graphe génère alors une combinatoire de configurations possibles. Au delà d'une certaine cadence de reproduction, la fréquence de reconception finit par imposer le recours au principe premier de préservation des invariances par séparabilité des éléments variables (pour éviter d'avoir à redéfinir intégralement les objets). Corrélativement s'impose le second principe de constructibilité des invariances par conception d'un langage des dénominations extensible aux identités conditionnelles et relatives. En somme, des cadences de reproduction et de renouvellement croissantes finissent par générer des objets subdivisibles et reconfigurables, nécessitant une stratégie de gestion des relations adaptée (avec variantes, versions, applicabilité, outils de génération, etc.) relayée par une stratégie de gestion de dénominations absolues ou bien relatives et conditionnelles (interchangeabilité, modularité, intégration continue, etc.). Les stratégies de gestion des relations et des dénominations doivent donc demeurer en adéquation avec les dynamiques industrielles soutenues par l'innovation et la sériation, sous peine d'un accroissement des risques, d'une multiplication des coûts et d'un allongement des délais.

Conclusion

Nous nous sommes attaqués au problème de la gestion des évolutions des données techniques dans les contextes de sériation et de forte évolutivité des objets ainsi que des systèmes de connaissances associés (sur lesquels se fondent les représentations formelles). Nous avons montré que la problématique de la gestion des impacts multiples et répétés sur les configurations pose des difficultés théoriques et pratiques qui ne peuvent être solutionnées de manière automatisée en conjuguant les fonctionnalités offertes par les P.L.M. La grande variabilité des contenus (compositions) et des contextes (configurations d'emploi) produisent des objets polymorphes aux

74 L'extension de marque par exemple en tant qu'instrument de différenciation des produits. "The brand, according to "American Marketing Association" is a name, term sign, symbol, design, or a combination of these purposed to identify the goods or services of one seller or group of sellers and so differentiate them from those of competitors." (Ladipo P.K.A., Olufayo T.O. et Omoera C.I., (2012), The Multi-Dimensional Application/Use of Branding in the Universe of Marketers, *International Journal of Academic Research in Business and Social Sciences*, Vol.2, N°4,

75 Les modifications proviennent essentiellement des améliorations (qualité, productivité, etc.) et des innovations.

76 Il est possible de considérer qu'il s'agisse d'une production matérielle ou immatérielle, d'objets, ou de définitions d'objets, ou de connaissances, etc.

77 Avec des cycles longs, plusieurs générations d'objets sont destinées à se côtoyer s'il y a renouvellement des définitions. Les objets à usage unique ou sans maintenance ont un cycle de vie réduit au cycle de production.

78 Dans toutes ses dimensions, enregistrement, stockage, recherche, accès, sécurité, etc.

79 Englobant les éventuels nouveaux éléments générés en vertu du premier principe de séparabilité des éléments variables.

identités paradoxales. Les stratégies et les formalismes doivent donc pouvoir, en fonction des dynamiques industrielles rencontrées, proposer une variabilité séparée ou simultanée des identités et des relations. Ces problématiques touchent les fondements conceptuels de l'identité et de la discernabilité, de l'unique et du multiple, des évolutions et des invariances. Elles se retrouvent aussi bien dans des domaines du génie logiciel que des systèmes de gestion des bases de données. Nous avons proposé une méthode automatisable⁸⁰ de traitement de ces objets polymorphes, au travers d'une conception nouvelle de l'identité des objets, relative et conditionnelle, permettant de traduire les équivalences de compositions en fonction des équivalences de configurations d'emplois. Nous avons également proposé des principes généraux pour mettre en correspondance les stratégies possibles de gestion des évolutions de configuration en fonction des situations industrielles génériques conjuguant de manière variable reconception (innovation) et reproduction (sériation). Ce travail mériterait bien entendu d'être prolongé dans plusieurs voies, notamment:

- ❖ Celle de la mise en œuvre instrumentale de la méthode proposée. Nous avons déjà bien avancé sur cette voie sur un terrain de l'industrie spatiale et de la santé notamment (Giacomoni, 2002; Giacomoni et Sardas, 2011). Cette expérimentation a permis de valider les concepts ainsi que leur implémentation informatique opérationnelle. Il resterait à travailler avec ou à partir des P.L.M actuels (les principales solutions du marché ont été examinées) pour étudier comment cette méthode pourrait s'intégrer à l'offre de fonctionnalité et sans doute aussi avancer dans une démarche de choix et d'articulation des différentes fonctionnalités.
- ❖ Celle de la réflexion et de l'expérimentation de la mise en œuvre de la démarche, du point de vue des enjeux organisationnels. Toute modélisation intégrée à un système d'information ne vaut qu'en fonction des compétences et des besoins des utilisateurs, qui sont ici multiples. Il nous faut distinguer deux catégories d'acteurs:
 - les administrateurs des données techniques, experts qui paramètrent le système à partir des fonctionnalités offertes. Il faut s'assurer que ces fonctionnalités correspondent non seulement à leurs besoins mais également à leurs modes de raisonnement, quitte à ce que ces derniers puissent évoluer dans le cadre de formations à de nouvelles démarches.
 - les différents acteurs opérationnels, concepteurs, fabricants, etc. qui doivent s'approprier la logique d'ensemble ainsi que les principes de modélisation sous-jacents, pour interpréter correctement leur rôle de pourvoyeur d'information et de validation des résultats produits par le système.

Ainsi au total, on pourrait avancer sur les enjeux de concrétisation sur les plans instrumental et organisationnel, en gardant bien à l'esprit que l'automatisation bien pensée doit faire une place adéquate aux experts tant pour le design des applications que pour la maîtrise de leur fonctionnement.

⁸⁰ Appliquée comme telle dans l'entreprise étudiée (spatial, défense, aéronautique)

BIBLIOGRAPHIE

- [1]. Abramovici M., (2007), "Future Trends in Product Lifecycle Management", in *The Future of Product Development*, Springer Berlin Heidelberg,
- [2]. Amann K., (2002), "Product lifecycle management: empowering the future of business", CIM Data, Inc.
- [3]. Batenburg R., Helms R. et Versendaal J., (2005), "The maturity of product lifecycle management in Dutch organizations: A strategic alignment perspective", *PLM'05: International conference on product life cycle management*, Lyon, France,
- [4]. Bellagio D. E. et Milligan T. J., (2005), *Software Configuration Management Strategies and IBM*
- [5]. Benbya H. et Meissonier R., (2007), "La contribution des Systèmes de Gestion des Connaissances au développement de nouveaux produits: étude de cas d'une entreprise du secteur de l'industrie aéronautique". *Systèmes d'Information et Management*, Vol.12, N°1,
- [6]. Bernard A., (1996), *Contribution à la modélisation des produits et à l'intégration des technologies productrices dans un environnement multi-acteurs*, HDR, Université Nancy I.
- [7]. Bertrand J.E., (1787), *Descriptions des arts et métiers, faites ou approuvées par Messieurs de l'Académie Royale des sciences de Paris. Avec figures en taille-douce*. Nouvelle édition, tome XV, Neuchâtel,
- [8]. Boothroyd G. & Dewhurst P., (1983), "Design for Assembly", University of Massachusetts,
- [9]. Bouikni N., Rivest L. et Desrochers A., (2008), "A Multiple Views Management System for Concurrent Engineering and PLM", *Concurrent Engineering: Research and Applications*, Vol.16, N°1,
- [10]. Bourdon I. et Lehmann-Ortega L., (2007), "Systèmes d'information et innovation stratégique: une étude de cas", *Systèmes d'Information et Management*, Vol.12, N°1,
- [11]. Bridges D. & Reeves S., (1999), "Constructive mathematics, in theory and programming practice", *Philosophia Mathematica*, Vol.7, N°1,
- [12]. Brönnimann H., Fabri A., Giezeman G.J., Hert S., Hoffmann M., Kettner L., Pion S., & Schirra S., (2009), *CGAL User and Reference Manual: All Parts*, Release 3.5, October,
- [13]. Bronsvoort W.F. et Noort A., (2004), "Multiple-view feature modeling for integral product development", *Computer-Aided Design*, Vol. 36,
- [14]. Brown J., (2006), "Managing Product Relationships: Enabling Iteration and Innovation in Design" *Business Value Research Series*, June, AberdeenGroup.
- [15]. Brunet J., (1991), "Modeling the World with Semantic Objects", in *Object Oriented Approach in Information Systems*, Van Assche F., Moulin B. & Rolland C. (Ed.), Elsevier Science Publishers B.V. (North Holland),
- [16]. Cantor G., (1883), *Mathematische Anllalen*, XXI, MiIner J.C. (tr.),
- [17]. Cantor G., (1895), "Beiträge zur Begründung der transfiniten Mengenlehre", in *Mathematische Annalen*, vol. 46,
- [18]. Carbonel M., (2001), "Dérives organisationnelles dans les projets ERP: les cas de Guerbet et Gaumont", *Systèmes d'Information et Management*, Vol.6, N°1,
- [19]. Chandler, A.D. Jr., (1977), *The Visible Hand: The Managerial Revolution in American Business*, Cambridge, Harvard University Press,
- [20]. Changeux J.-P. et Connes A., (1989), *Matière à pensée*, Odile Jacob (ed),
- [21]. Chapell W., (1970), *A Short History of the Printed World*, Hartley & Marks Publishers Inc., Canada,
- [22]. CIM Data, (2002), "Étude de l'offre SAP pour le cPDM: my SAP PLM, Pilotage des nouveaux produits et processus grâce à la gestion du cycle de vie des produits." SAP.AG, Allemagne,
- [23]. Codd E.F., (1990), "The relational model for database management" version 2, by Addison-Wesley Publishing Company, Inc.

- [24]. Codd E.F., (1970), "A Relational Model of Data for Large Shared Data Banks" CACM 13, N°6, June, Information Retrieval P. Baxendale, Ed.,
- [25]. Cohen Y., (1994), "Inventivité organisationnelle et compétitivité: l'interchangeabilité des pièces face à la crise de la machine-outil en France autour de 1900", *Entreprises & Histoire*, N°5, juin,
- [26]. Demoly F., (2010), *Conception intégrée et gestion d'informations techniques: application à l'ingénierie du produit et de sa séquence d'assemblage*, Thèse, Université de Technologie de Belfort-Montbéliard, Gomes S. (Dir.)
- [27]. Dirickersbach J.T., (2008), *Supply Chain Management with SAP APO*, Springer,
- [28]. Djezzar L., (2003), *Gestion de configuration*, Dunod, Paris,
- [29]. Eicher C. et al., (1984), *Mise en oeuvre et réalités de la Gestion de Production Assistée par Ordinateur*, Editions du CESTA,
- [30]. Fodor X., (2008), "Organiser les échanges de données techniques: ARIANESPACE ne travaille pas dans la précipitation", Dossier: Collaboration: Optimisez votre travail, *iTechnologie*, N°2.
- [31]. Feeney M., (1999), *Digital Culture: Maximising the Nation's Investment: a Synthesis of JISCO/NPO Studies on the Preservation of Electronic Materials*, Londres, National Preservation Office,
- [32]. Fine K., (2002), *The Limits of Abstraction*, Oxford University Press,
- [33]. Frege, G. (1994), *Ecrits logiques et philosophiques*, trad. Imbert C., Ed. du Seuil,
- [34]. Frege G., (1879), *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*, Halle a.S.: Nebert L., Bauer S. Mengelberg (tr.), (1879-1931), *Concept Notation: A formula language of pure thought, modelled upon that of arithmetic*, in van Heijenoort J., *From Frege to Gödel: A Sourcebook in Mathematical Logic*, Cambridge, MA: Harvard University Press,
- [35]. Frege, G., (1884), *Die Grundlagen der Arithmetik: eine logisch-mathematische Untersuchung über den Begriff der Zahl*, Breslau: Koebner w., Austin J.L. (tr.), (1974), *The Foundations of Arithmetic: A Logic-Mathematical Enquiry into the Concept of Number*, Oxford: Blackwell, 2nd ed.,
- [36]. Frege G., (1893-1903), *Grundgesetze der Arithmetik*, Band I/II, Jena: Verlag Pohle H., Furth M. (tr.), (1964), *The Basic Laws of Arithmetic*, Berkeley: U. California Press, vol.I/II,
- [37]. Geach P., (1980), *Logic Matters*, University of California Press,
- [38]. Ghouil S., (1983), *Base de données et gestion de configurations dans un atelier de génie logiciel*, I.N.P., Grenoble, Thèse de doctorat d'ingénieur,
- [39]. Giacomoni G., (2002), "Gérer la reproduction d'objets complexes dans un contexte d'innovation permanente: le cas de l'industrie de l'espace", Thèse de Doctorat, ENSMP, Paris,
- [40]. Giacomoni G. et Sardas J.-C., (2011), "Pilotage des productions d'objets complexes dans l'industrie de l'Espace : innovation intensive et sériation", *Revue Française de Gestion Industrielle*, Vol.30, N°1,
- [41]. Giard V., (2003), *Gestion de la production et des flux*, 3e édition, Economica,
- [42]. Girard J.Y., (2009), "Identité, égalité, isomorphie; ou ego, individu, espèce", d'après un exposé à la réunion LIGC opus 10, Firenze, villa Finaly, 18 Septembre 2009.
- [43]. Godement R., (2001), *Analyse mathématique*, Vol.1. Convergence, fonctions élémentaires 2^{ème} édition corrigée, Springer-Verlag, Berlin, Heidelberg New York,
- [44]. Gomes S. et Sagot J.-C., (2002), "A concurrent engineering experience based on a cooperative and object oriented design methodology" in Best papers book from 3rd *International Conference on Integrated Design and Manufacturing in Mechanical Engineering*, Kluwer Publishers,
- [45]. Grieves M., (2006). *Product Lifecycle Management: Driving the Next Generation of Lean Management.*, New York, NY: McGraw Hill,
- [46]. Hatchuel A., Sardas J.C. et Weil B., (1988), "La mise en oeuvre et le pilotage d'une GPAO: à chaque étape ses difficultés", *Revue de l'AFGI*,
- [47]. Hatchuel A. et Sardas J.C., (1992), "Les grandes transitions contemporaines des systèmes de production: une approche typologique", in Dubois M., de Terssac G. (dir.), *Les nouvelles*

rationalisations de la production, CEPADUES.

- [48]. Hatchuel A. & Weil B., (2007), "Design as Forcing: deepening the foundations of C-K theory", Paper submitted at ICED 07, Paris,
- [49]. Howse D., (1980), *Greenwich Time and the Discovery of the Longitude*, Oxford University Press; 1st Ed.,
- [50]. Huang Q., (1996), *Design for X: concurrent engineering imperatives*, Chapman & Hall (Ed.),
- [51]. Hussenot A., (2007), "Dynamiques d'appropriation organisationnelle des solutions TIC: une approche en termes de démarches itératives d'appropriation", *Systèmes d'Information et Management*, Vol. 12, N°1,
- [52]. Hwang J., Mun D. et Han S., (2009), "Representation and Propagation of Engineering Change Information in Collaborative Product Development using a Neutral Reference Model", *Concurrent Engineering: Research and Applications*, Vol.17, N°2,
- [53]. IBM, Dassault Systèmes, (2008), "IBM and Dassault Systèmes: Business Process Accelerators for Systems Engineering □ Integrated Product Development from needs identification through to final product validation".
- [54]. Jech T., (1978), *Set theory*, Academic Press,
- [55]. Jerdack G.R. et Pranab K.S., (1990), □Annals of the Institute of Statistical Mathematic□ Volume 42, Number 1/mars 1990, Springer Netherlands,
- [56]. Kosman L.A., (1969), □Aristotle's Definition of Motion□ *Phronesis*,
- [57]. Lawrence P. et Lorsch J., (1967), *Adapter les structures de l'entreprise - Intégration ou différenciation*, trad. fr., Paris, Editions d'organisation,
- [58]. Lee S.P., Brunet J. & Rolland C., (1992), "Abstraction in the O* Object-Oriented Method", IFCPAR, Proceedings Indo-French Workshop on Object-Oriented Systems, nov., Goa India,
- [59]. Lehmann-Haupt H., (1966), *Gutenberg Master of the Playing Card*, New Haven, Yale University Press,
- [60]. Le Moigne J.-L., (1977), *La théorie du Système Général, théorie de modélisation*, P.U.F., Paris, 1977, 3^{ème} édition mise à jour.
- [61]. Leibnitz G.W., (1714), *Monadologie*, Trad. Parmentier M., Paris, Vrin,
- [62]. Leibnitz G.W., (1998), *Recherches générales sur l'analyse des notions et des vérités*, Epiméthée, PUF, Paris,
- [63]. Littré E., (1976), *Dictionnaire de la langue française*, tome III, éditions Famot, Genève,
- [64]. Mantripragada R. et Whitney D.E., (1999), "Modeling and Controlling Variation Propagation in Mechanical Assemblies using State Transition Models", *IEEE Transaction on Robotics and Automation*, Vol. 15, No. 1,
- [65]. Marciniak R. et Rowe F., (2008), *Systèmes d'information, dynamique et organisation*, Economica
- [66]. McMurtie Douglas C., (1942), *The invention of Printing. A Bibliography*, Rochester, N.Y.: Publishing And Printing Dept. Of The Rochester Athenaeum And Mechanics Institute For The Educational Commission Of The International Association Of Printing House Craftsmen,
- [67]. Merminod V., Mothe C. et Rowe F., (2009), "Effets de Product Lifecycle Management sur la fiabilité et la productivité: une comparaison entre deux contextes de développement produit." *M@n@gement*, 12(4),
- [68]. Merminod V., (2007), "TIC, Partage de connaissances et fiabilité du développement produit distribué: une approche par le "glitch" au sein du Groupe SEB", *Systèmes d'Information et Management*, Vol. 12, N°1,
- [69]. Micklem et al., (1975), □Compartments and cell flows within the mouse haemopoietic system. I. Restricted interchange between haemopoietic sites□ *Cell Tissue Kinet.*, May, Vol.8 N°3,
- [70]. Mostefai S. et Batouche M., (2005), "Data integration in Product Lifecycle Management: an ontology-based approach", PLM'05: International conference on product life cycle management, Lyon, France.

- [71]. Neagu N. Et Faltings B., (2001), "Exploiting Interchangeabilities for Case Adaptation" Lecture notes in computer science, Springer, Berlin, Allemagne, Etats-Unis,
- [72]. Noël F., (2006), "A dynamic multi-view product model to share product behaviors among designers: how process model adds semantic to the behavior paradigm", *International Journal of Product Lifecycle Management*, Vol.1, N°4,
- [73]. Patout Y., (2001), "La réorganisation de la téléphonie par les profils", *Systèmes d'Information et Management*, Vol. 6, N°1,
- [74]. Perrow C., (1967), "A Framework for the Comparative Analysis of Organizations", in *American Sociological Review*, Vol.32, N°2,
- [75]. Pol G., Jared G., Merlo C. et Legardeur J., (2005), "From PDM systems to integrated project management systems: a case study", *PLM'05: International conference on product life cycle management*, Lyon, France.
- [76]. Ramsden J., (1787), *Description d'une machine pour diviser les instruments de mathématiques*, Londres, Bureau des Longitudes, traduite de l'Anglais, Paris,
- [77]. Raymond E.S., (1998), *The Cathedral and the Bazaar*, O'Reilly,
- [78]. Rehman F.U., Yan X.-T., (2007), "Supporting early design decision making using design context knowledge", *Journal of Design Research*, Vol.6, N°1-2,
- [79]. Reix R., Kalika M., Fallery B. & Rowe F., (2011), *Systèmes d'information et management des organisations*, 6e éd., Vuibert,
- [80]. Rice H.G., (1953), "Classes of recursively enumerable sets and their decision problems", *Trans. Am. Math. Soc.*, vol. 74, n°2,
- [81]. Rousseau F., (2005), *Conception des systèmes logiciel/matériel: du partitionnement logiciel/matériel au prototypage sur plateformes reconfigurables*, Thèse d'HDR, Université Joseph Fourier - Grenoble I,
- [82]. Russel B., (1900), *A Critical Exposition of the Philosophy of Leibnitz*, Cambridge, The University Press,
- [83]. Russel B., (1903), *The principles of mathematics*, New York,
- [84]. Saga V. et Zmud R. W., (1996), « Introduction de logiciels de gestion dans des petites entreprises liées à une profession libérale », *Systèmes d'Information et Management*, Vol.1, N°1,
- [85]. Sacquet A. et Nowencien R., (1995), "Productivité de la maintenance en Architecture client-serveur", revue *Génie Logiciel* n°35,
- [86]. Scholderer V., (1970), *Johannes Gutemberg, The Inventor of Printing*, British Museum, 2nd Ed.,
- [87]. Simon H., (1976), "From substantive to procedural rationality", in: S. LATSIS, *Method and appraisal in economics*, Cambridge, Cambridge University Press,
- [88]. Simondon G., (1958), *Du mode d'existence des objets techniques*, Aubier,
- [89]. Stark J., (2004), "Product Lifecycle Management - 21st century Paradigm for Product Realization.", *Decision Engineering Series*, Springer Verlag, Berlin,
- [90]. Steiner G. et Stephenson P., (2000), "Subset-Restricted Interchange for Dynamic Min-Max Scheduling Problems" *SIAM Journal on Discrete Mathematics*, archive Vol.13, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA,
- [91]. Turing A.M., (1936), "On Computable Numbers, with an Application to the Entscheidungs problem", *Proceedings of the London Mathematical Society*, Series 2, 42,
- [92]. Turing A.M., (1948), "Intelligent Machinery", National Physical Laboratory Report, in Meltzer B., Michie D., (eds), (1969), *Machine Intelligence*, 5, Edinburgh University Press, Edinburgh,
- [93]. Updike D.B., (1920), *Printing Types: Their History, Forms and Use*, 2 vol., Harvard and Oxford,
- [94]. Wiggins D., (2001), *Sameness and Substance Renewed*, Cambridge University Press,
- [95]. Woodward J., (1965), *Industrial organization. Theory and practice*, Oxford University Press,

Annexe 1 : Extraits des descriptifs fonctionnels

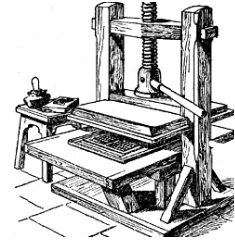
<p>¹ <input type="checkbox"/> <input type="checkbox"/> the potential for re-use physical items determined by using the names and descriptions to identify the similarities and differences between items naming conventions <input type="checkbox"/> a generic noun is used as the name to describe each physical item <input type="checkbox"/> each item is described by identifying its attributes in their descending order of significance [Manufacturer part number] <input type="checkbox"/> form of interchangeability codes (EDI standards) for restricted additional interchangeability relationships <input type="checkbox"/> user interface <input type="checkbox"/> serial number <input type="checkbox"/> indices de modification d'article <input type="checkbox"/></p> <p><input type="checkbox"/> <input type="checkbox"/> numéro de modification <input type="checkbox"/> versions <input type="checkbox"/> variantes <input type="checkbox"/> <input type="checkbox"/></p> <p><input type="checkbox"/> <input type="checkbox"/> Interchangeability of similar items is confirmed by comparing their designs <input type="checkbox"/> full interchangeability form-fit-function-class (FFF class) <input type="checkbox"/> restricted interchangeability relationships between groups of fully interchangeable parts <input type="checkbox"/> sub-item for company-internal technical mapping of a part interchange <input type="checkbox"/> définition et gestion de hiérarchies de familles d'articles pour la recherche des articles standard <input type="checkbox"/></p> <p><input type="checkbox"/> <input type="checkbox"/> extension d'une division articles <input type="checkbox"/> classification <input type="checkbox"/> article configurable vs non-configurable <input type="checkbox"/> dépendances de nomenclatures <input type="checkbox"/> nœud (qui représente un composant ou une fonction de la structure produit) <input type="checkbox"/> schémas de variance pour vérifier la complétude et la non ambiguïté des variantes (cas où plusieurs solutions sont générées pour un seul choix de variante) <input type="checkbox"/></p> <p><input type="checkbox"/> <input type="checkbox"/> inventory overview for the stocks of interchangeable parts in context <input type="checkbox"/> cumulative analysis of interchangeable parts <input type="checkbox"/> ERP links <input type="checkbox"/> La fonctionnalité avancée [X] insère les processus de gestion des modifications entre l'ingénierie et la fabrication: analyse des modifications induites à la recherche de conflits; génération d'un état signalant les avertissements, les erreurs ou l'absence de conflit; recherche des ordres de fabrication impactés, création des demandes de modification correspondantes <input type="checkbox"/> <input type="checkbox"/></p>	[SAP <input type="checkbox"/> PLM]
<p><input type="checkbox"/> <input type="checkbox"/> standard <input type="checkbox"/></p> <p><input type="checkbox"/> <input type="checkbox"/> articles sérialisés <input type="checkbox"/> cas d'emploi et rechanges <input type="checkbox"/> <input type="checkbox"/></p> <p><input type="checkbox"/> <input type="checkbox"/> suivi et traçabilité <input type="checkbox"/> analyse d'impacts <input type="checkbox"/> configurations de référence (baselines) <input type="checkbox"/> variantes et effectivités à date ou rang <input type="checkbox"/> historique des versions et comparaison des configurations <input type="checkbox"/> traitement des anomalies et des dérogations <input type="checkbox"/></p> <p><input type="checkbox"/> <input type="checkbox"/> chaîne de décision <input type="checkbox"/> interface avec l'ERP <input type="checkbox"/> processus de pré-approvisionnement et traitement des modifications de configurations en lien avec les contraintes des achats et de la production <input type="checkbox"/> <input type="checkbox"/></p>	[LASCOM <input type="checkbox"/> ICS]
<p>¹⁰ <input type="checkbox"/> <input type="checkbox"/> standard <input type="checkbox"/></p> <p>¹¹ <input type="checkbox"/> <input type="checkbox"/> generic product, platform, variant requirements: systematic approach to organize products (logical set of modules) into interchangeable modules <input type="checkbox"/> appropriate module interfaces <input type="checkbox"/> <input type="checkbox"/></p> <p>¹² <input type="checkbox"/> <input type="checkbox"/> baselines, release level, effectivity <input type="checkbox"/> <input type="checkbox"/></p> <p>¹³ <input type="checkbox"/> <input type="checkbox"/> translation of requirements into required functions <input type="checkbox"/> high-level platform architecture design <input type="checkbox"/> product module interfaces formalized to manage the transfer and reintegration of design data, while preserving its integrity <input type="checkbox"/> changes control option sets for Assemble-to-Order products <input type="checkbox"/> options; combinations <input type="checkbox"/> <input type="checkbox"/></p> <p>¹⁴ <input type="checkbox"/> <input type="checkbox"/> decision management process; formal change impact and root cause analysis; configuration traceability <input type="checkbox"/> <input type="checkbox"/></p>	[PTC - Windchill]
<p>¹⁵ <input type="checkbox"/> <input type="checkbox"/> serial numbers units <input type="checkbox"/> alternate parts and acceptable substitutions are communicated and managed to allow for manufacturing flexibility with adequate control for managing quality standards <input type="checkbox"/> <input type="checkbox"/></p> <p>¹⁶ <input type="checkbox"/> <input type="checkbox"/> re-use of multiple product variants or product evolution; tracking original requirement to the final product <input type="checkbox"/> <input type="checkbox"/></p> <p>¹⁷ <input type="checkbox"/> <input type="checkbox"/> modifications for a set of configurations <input type="checkbox"/> specific bills-of-material views for unique product and/or tracking changes; product configuration alternatives with traceability for design changes, analysis of impacts, detail design, downstream processes and compatible effectivity (...) configured infrastructure linking product design with logical and functional definitions; configured context for product components design and modifications; effectivity ranges for added and modified product components; variants or state of the product <input type="checkbox"/> <input type="checkbox"/></p> <p>¹⁸ <input type="checkbox"/> <input type="checkbox"/> a key requirement for engineering change is the approval and notification process <input type="checkbox"/> viewing and tracking of all changes to all product configurations; program deliverables with reliable and up-to-date information to all stakeholders at all times; investigation of new variation of product with virtual testing and performance-based decision-making; risk management <input type="checkbox"/> <input type="checkbox"/></p> <p>¹⁹ Cette solution se base sur les produits logiciels CATIA, ENOVIA, DELMIA, SIMULIA</p>	[DASSAULT Systèmes - DS Portfolio]
<p>²⁰ Teamcenter, NX, Tecnomatic, Velocity/Solid Edge</p> <p>²¹ <input type="checkbox"/> <input type="checkbox"/> impacts of a design change managing the relationships between parts, their technical data, and the documentation that supports them <input type="checkbox"/> data assembly code; identification/status information which includes various kinds of metadata that determines the access rights and configuration controls for the data module; model identification <input type="checkbox"/> <input type="checkbox"/></p> <p>²² <input type="checkbox"/> <input type="checkbox"/> analyse d'impacts; gestion du contexte; synthèse/comparaison pour le contrôle de l'assemblage par rapport à la conception; comparaison des différentes nomenclatures à l'état d'origine; traçabilité des éléments critiques <input type="checkbox"/> implied relationships between product structure and parts data (relative to part numbers, configuration and effectivity) <input type="checkbox"/></p> <p>²³ <input type="checkbox"/> <input type="checkbox"/> standard numbering system; proven interoperability through a common information platform for integration [content interchangeability and use of context-driven content]; relationships within a subsystem and across the rest of the platform subsystems [to ensure the primary system complies with the most challenging requirements (from concept development through production)]: managing the inaction and impact of all subsystems and ultimately ensuring the collective performance of all subsystems satisfies the overall platform/prime system/vehicle performance requirements <input type="checkbox"/> commonality and re-use capabilities <input type="checkbox"/> configuration and change management with digital simulation <input type="checkbox"/> <input type="checkbox"/></p> <p>²⁴ <input type="checkbox"/> <input type="checkbox"/> modular scenario for data management {content/identification + status data (metadata)} necessary for controlling the data module and its configuration; design process and variants module enhancements (links between variant filters and variant sets in design process) <input type="checkbox"/> <input type="checkbox"/></p> <p>²⁵ <input type="checkbox"/> <input type="checkbox"/> fully understand the impact of product changes; collaborative solution for planning and validating the manufacturing assembly processes; evaluate different assembly alternatives, plan for multiple variants and management change across the entire assembly process lifecycle; review and adjustment of the product; multiple decisions about what elements and attributes of the specification need to be used; multiple lifecycle states <input type="checkbox"/> <input type="checkbox"/></p>	[SIEMENS <input type="checkbox"/> Portfolio]

Annexe 2 : Machine à imprimer (interchangeabilité de seconde espèce) et horlogerie (interchangeabilité de première espèce)


Fig.5 - Machine à imprimer - Interchangeabilité de seconde espèce

La méthode de fabrication des caractères moulés:

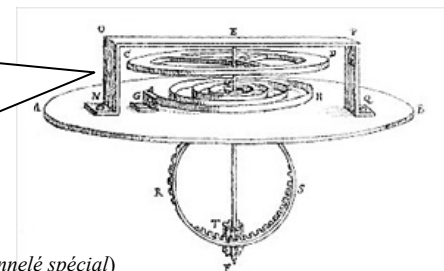
- *gravure* en poinçon à l'extrémité d'une tige métallique en acier doux, du signe voulu en relief et à l'envers
- *frappe* à l'aide de ce poinçon d'une matrice dans un métal plus tendre (cuivre rouge)
- *moulage* (reproduction), dans un moule à arçon (à main), d'un alliage ternaire (plomb, étain, antimoine)
- *disposition* (mots, phrases, pages, □) des *types* sur un socle en les mettant à l'envers
- *disposition* d'une feuille de papier dessus
- *compression* forte d'une platine sur un bloc d'impression parfaitement plan
- *impression* de pages recto verso et reproduction à l'identique.



(Nouvelle machine outil □ *machine à diviser*)



(nouvel élément / *vis-mère*)



(nouvel élément : *cône cannelé spécial*)








Fig.6 - Horlogerie □ Interchangeabilité de première espèce

Emergence-Diffusion de l'innovation:
1^{ère} espèce [réemploi - cas d'emploi]

- ❖ objet modifié (*horloge à ressort*) ou différent (*instruments scientifiques: télescope, sextant, microscope, □*)
- ❖ élément(s) invariant(s) (*de l'horloge à ressort*) ou diffusés (*vis-mère, ressort, roue dentée, □*)

(nouvel objet - *horloge à ressort*;
Innovation: un nouveau système moteur (ressort))