

**Ecole Doctorale
Organisation Industrielle et Systèmes de Production
(O.I.S.P.)**

Doctorat de Génie Industriel
Institut National Polytechnique de Grenoble

Résumé de Thèse

***Proposition de mécanismes de versionnement et
d'agrégation des connaissances de domaine en
conception collaborative de produit industriels***

Par Alaeddine ZOUARI

Laboratoire G-SCOP (Sciences pour la Conception, l'Optimisation et la Production de Grenoble)

Ecole Doctorale « Organisation Industrielle et Systèmes de Production »

Directeur : Daniel BRISSAUD - Assistante : Christelle GRANDVALLET

BP 47 – 38 040 Grenoble Cedex 9

Mail : ed-oisp@inpg.fr Tel : 04 76 82 56 82

Soutenance le 02/03/2007 à Sfax – N° XX



Alaeddine est né le 02 Aout 1966 à Sfax-Tunisie (3000). Il est titulaire d'un Baccalauréat math technique, d'une Maîtrise es science technique option construction mécanique, d'une certéficat d'aptitude à l'enseignement technique (ENSET Tunis), d'un diplôme des études supérieures spécialisé en Génie Mécanique (ESST Tunis) puis d'un Diplôme d'Etudes Approfondies en Mécanique et ingénierie (ENI Sfax). Il a opté pour une cotutelle de thèse en Génie Industriel entre l'INPG et L'université de Sfax-Tunisie au sein du laboratoire GILCO : Gestion Industrielle, Logistique et Conception et du laboratoire (LASEM) Laboratoire des Eystèmes ElectroMécaniques.

Il a commensé sa vie active en 1992 comme professeur d'enseignement technique au lycée Zéramdine-Monastir, puis comme assistant technologue à l'Institut Supérieur des Etudes Technologiques de Sfax entre 1998-2004 et depuis septembre 2004 il est assistant universitaire à l'Institut Préparatoire aux Etudes d'Ingénieurs Elmanar-Tunis.

Thèse soutenue le 02 Mars 2007

Directeur de thèse :

Michel TOLLENAERE	Professeur à l'INPG
Aref Younes MAALEJ	Professeur à l'ENIS

Codirecteur de thèse :

Habib BEN BACHA	MDC à l'ENIS
-----------------	--------------

Jury :

Chedly BREDAI	(ENIS-Université de Sfax)	, Président
Benoît EYNARD	(UTT-Université de Troy)	, Rapporteur
Anis CHELBI	(ESSTT-Université de Tunis II)	, Rapporteur
Wassila BOUZID SAI	(ENIS-Université de Sfax)	, examinatrice
Michel TOLLENAERE	(ENSGI-INPG)	, directeur de thèse
Aref Younes MAALEJ	(ENIS-Université de Sfax)	, directeur de thèse
Habib BEN BACHA	(ENIS-Université de Sfax)	, codirecteur de thèse

Proposition de mécanismes de versionnement et d'agrégation des connaissances de domaine en conception collaborative de produit industriels

I Introduction

La conception est une activité complexe. Cette complexité est due pour une grande part à la position qu'occupe la conception au sein des entreprises et aux nombreux acteurs et organisations qui y interviennent. Elle doit aujourd'hui répondre à différents objectifs, porteurs de multiples contraintes. Toutefois la conception ne se limite pas à une activité technique ou scientifique. Elle constitue également pour les entreprises une activité économique car l'acte de concevoir sert, en outre, à produire une valeur ajoutée, indispensable pour assurer la pérennité des entreprises. Dans le cadre de nos travaux, nous nous intéressons principalement à la conception routinière multi acteurs. Dans cet environnement de conception collaborative, la connaissance peut être capitalisée dans une base de données (mémoire de groupe ou mémoire de projet) [Ribière et al. 1998]. Cette base de données permet à partager la connaissance entre les acteurs impliqués dans le processus de conception, et peut servir de référentiel de connaissance aux entrées ou à la sortie du processus de tâches.

Nos travaux se sont basés sur les modèles pour la capitalisation et la réutilisation des connaissances en conception routinière dans un contexte de co-conception (projet MULTI thèse de Sébastien Menand [Menand, 2002]). Dans une première étape, nous avons commencé par la structuration et l'affinage des modèles proposés afin de faciliter leur mise en place en base de données Access ou Oracle dans l'intention de développer une méthodologie qui a pour objectif d'aider les acteurs impliqués dans le processus de conception, à gérer les connaissances et les données techniques au cours de leur travail. Pour valider ces concepts, ils ont été appliqués sur le domaine des systèmes de climatisation automobile.

Le deuxième volet consiste à poursuivre ces travaux à travers l'élaboration d'une méthode qui met en émergence la collaboration des acteurs impliqués dans un processus de conception en vue de construire dynamiquement un référentiel de connaissances, qui trace l'historique du processus d'ingénierie et capitalise les connaissances et les ressources liées au produit dans le but de réutiliser ce patrimoine informationnel. L'objectif est de fournir aux acteurs pilotes un outil de conduite technique des projets, capable de supporter le recueil de tous les points de vue, leur traduction en termes techniques sur l'objet de conception, puis leur intégration dans le processus. Pour faciliter les tâches des concepteurs impliqués dans un projet de conception routinière, dans un contexte d'ingénierie concourante, une bibliothèque de connaissances et un outil d'ingénierie basé sur l'information (KBE : knowledge based engineering) s'avèrent nécessaires. Cet outil sera utilisé comme un support de flux d'informations fonctionnelles partageables entre les différents acteurs pouvant travailler sur différents sites et différents projets.

La partie cruciale de cette étude consiste à étudier de façon formelle les mécanismes d'agrégation de modèles de connaissances de domaine. Le domaine des connaissances est continuellement modifié par des acteurs différents, ce qui implique que sur le référentiel de connaissances nous allons voir apparaître de nouvelles connaissances (actualisation, enrichissement...) et des modifications sur les anciennes (expansion, restriction, changement...). Ceci va engendrer une grande diversité de connaissances, difficile à gérer. Pour résoudre ce problème, tout en gardant les traces des anciens modèles de connaissances nous proposons d'étudier les versions de connaissances. Les connaissances capitalisées sont des connaissances de métier (connaissances techniques) associées au domaine de l'application dont l'exploitation est possible par l'intermédiaire d'un outil de partage d'informations et permettant leur consultation. Cependant, l'analyse sémantique exige des outils informatiques pour comprendre la structure et le lexique de cette connaissance, de ce fait rendre leur implication explicite (vérification) et détecter les éventuelles contradictions (validation) [Sellini, 1999].

Le versionnement consiste à étudier la capacité de contrôler les modifications et les conséquences requises et avoir une traçabilité du processus de conception (sous forme de versions). Les recherches sur le versionnement des connaissances sont très rares. C'est une fonctionnalité critique dans ce cadre de traiter le versionnement et la gestion des connaissances multiples. En fait, le versionnement nécessite une analyse complète des connaissances disponibles, qui est basée sur les modifications présentées par la transformation d'une version de connaissance en une autre différente. Une telle analyse mène à la spécification de la traçabilité de l'évolution des versions de connaissances qui permet de les capitaliser, en évitant d'éventuelles redondances, dans le but de les réutiliser sur d'autres projets. Pour pallier ces problématiques, nous proposons un mécanisme permettant d'exprimer des règles d'évolution et de définir et gérer des stratégies d'évolution. Les règles d'évolution proposées ici permettent d'exprimer les vérifications et les propagations de mise à jour à exécuter avant et après une opération faisant évoluer une version de connaissance.

L'étude a conduit au développement d'une maquette informatique appelé «DISCOVER» (DeSIgn with KnOwledge VERsioning). Le modèle à trois niveaux de connaissance affiné y est implémenté afin de permettre son "remplissage" avec des connaissances, informations et données d'un domaine particulier (par exemple : la conception des systèmes de climatisation dans notre cas d'application). Le prototype informatique permet par sa consultation et sa mise à jour de valider l'approche proposée.

II Modèle de connaissance

Le modèle que nous proposons vise à être générique à toute conception de produit. Il décrit le niveau générique des connaissances relatives à un produit et à son processus de conception et permet l'émergence des solutions par l'intégration des contraintes et des exigences révélant du cycle de vie du produit (phases et situation de cycle de vie). Il est en fait un ensemble de modèles allant du paramètre élémentaire à la notion de projet et permet de structurer l'information et les connaissances en trois niveaux. Le premier représentant le domaine de l'ingénierie système en conception fonctionnelle (modèle générique), le second hébergeant le domaine technique (référentiel de connaissance de domaine) et le troisième permettant de capitaliser au fil de l'eau et de distribuer les données instanciées dynamiquement au long des différents projets de conception (bibliothèque de projets) [Zouari et al, 2005].

Le modèle générique est apte, par exemple, à supporter le processus de développement selon l'approche du cycle en V de l'ingénierie système. Il décrit les connaissances selon les points de vue, produit et processus de conception, d'où la naissance d'un modèle produit et d'un modèle processus de conception. Le modèle produit illustre les entités relatives à un produit d'une manière générale tels que les paramètres, les articles, les sous systèmes, les caractéristiques, les parties, les contraintes...etc. Par analogie le modèle processus de conception présente celles relatives à un processus de conception tels que les tâches, les acteurs, les règles, les rôles, les conditions d'exécution...etc. [Menand, 2002].

Les modèles ainsi obtenus seront instanciés sur le domaine traité. Nous obtenons alors un modèle produit et un modèle processus de conception spécifique à un domaine technique particulier. Ce résultat constitue un référentiel de connaissance de domaine applicable à la conception de tout artefact de ce domaine.

L'instanciation des modèles de domaine (produit et processus de conception) sur un projet permet de concevoir un produit requis au sein d'un projet. Nous obtenons alors un référentiel de connaissance du projet. L'ensemble des référentiels de connaissances de projet constitue la bibliothèque de projet qui se compose de produits conçus et de processus de conceptions exécutés.

Le développement d'une telle mémoire correspond à une phase de la capitalisation entière de la connaissance. Ceci exige de fournir à chaque concepteur un point de vue adapté sur le projet et de traduire leurs recommandations dans la langue intelligible pour les concepteurs de DAO [Eckert et al, 2003]. Alors, si avec ces propositions, recommandations et évaluations, le concepteur travaillera dans la mesure du possible sur l'intégration de toutes les demandes du prochain projet. Une telle étape de capitalisation des connaissances place néanmoins des problèmes en termes d'identification, redondance et réutilisation de cette connaissance,

d'où la nécessité de procéder à la création de versions de connaissances (figure 1). Le modèle permet, entre autres, de formaliser le produit avec ses fonctions ainsi que les différentes situations de vie que celui-ci rencontre. Il permet alors de capturer les connaissances concernant la vie du produit ainsi que de garder une traçabilité des contraintes rencontrées en conception et des choix effectués en conséquence. Ces concepts seront tout particulièrement développés dans l'étude de part leur importance dans la conception de produit.

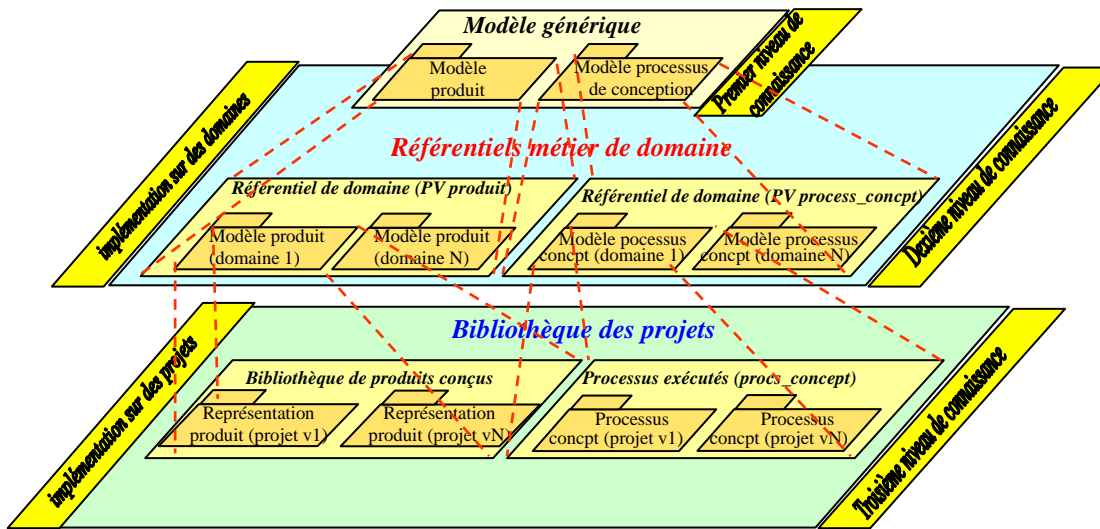


Figure 1. Les trois niveaux de connaissances

Les nouvelles technologies de l'information rendent aujourd'hui possible le réel partage d'un modèle produit ainsi que celui du modèle du processus de conception. Le type de conception plus particulièrement visé dans l'étude est la « conception routinière de produits ». Dans ce type de conception, les concepts technologiques existent déjà et sont bien connus des concepteurs qui n'ont plus qu'à les dimensionner et à les intégrer dans un nouvel environnement [Zouari et al. 2007]. L'objectif étant que les technologies choisies réalisent les performances requises par les clients en respectant les contraintes de l'environnement.

III La dynamique des connaissances dans les outils de conception

III.1 Nature dynamique de la conception

Un des problèmes principaux, qui doit être adressé en définissant les modèles informationnels pour la conception est provoqué par la nature dynamique de la conception. Pendant le processus de la conception, l'information n'est pas traitée en tant que données statiques, mais est sujette invariablement au changement. C'est dû au caractère de résolution des problèmes de la conception, qui implique la recherche de l'information, d'analyser, de structurer, d'interpréter, et d'évaluer l'information en répétant des cycles. Le plus important, il comporte la génération d'information pendant ce processus cyclique et la combinaison de cette nouvelle information avec des données connues, définissant les nouvelles structures d'information qui mènent aux solutions de la conception. Nous concluons que la définition et la structure de l'information pendant le processus cyclique de la conception est une question clé dans les tâches support du processus de conception avec des modèles informationnels et des systèmes de gestion de l'information de conception.

III.2 Ingénierie des connaissances en conception collaborative

La capitalisation des connaissances vise à la fois la préservation, le partage et surtout la réutilisation d'un savoir-faire généré au fil des projets de conception. L'effort de capitalisation passe plus concrètement par la modélisation ayant comme objectif de diminuer le temps de conception, en réutilisant ce qui est déjà validé, pour se concentrer uniquement sur les nouveaux choix à gérer pour une nouvelle solution. De nombreux auteurs ont proposé des modèles à base de connaissances visant

la conception collaborative : Modèle produit [Tollenaere, 1994], [Tichkiewitch, 1996], [Yvars, 2001], [Menand, 2002], etc. Modèle processus de conception [Dupinet, 1991], [Harani, 1997], [Ouazzani, 1999], etc.

Plusieurs projets servent de support au travail dit « en avance de phase » dans le domaine de l'ingénierie des connaissances tel que DEKLARE (Design Knowledge Acquisition and Redesign Environment) et MOKA (Methodology and tools Oriented to Knowledge Base Engineering Application project) IPPOP (modélisation intégrée produit-processus-organisation), etc. Le projet declare, ([Vargas, 1995], [Saucier, 1997], [Mahe, 1998] etc.) a permis de définir un modèle de représentation de connaissances de conception en ingénierie des systèmes mécaniques, s'intéressant plus particulièrement à la modélisation de problèmes relevant de la conception routinière. Il décrit le produit de plusieurs manières (point de vue physique, fonctionnel, géométrique et de contrainte), et le modèle du processus de conception comme étant une suite de tâches à effectuer pour concevoir un objet donné.

L'avènement des technologies de l'information ouvre de nouvelles perspectives de partage d'informations et d'expériences entre les différents acteurs dans une entreprise [Wallace et al. 1997], particulièrement les concepteurs travaillant sur un même projet. Dans un environnement de conception collaborative, la connaissance peut être capitalisée dans une base de données (mémoire de groupe ou mémoire de projet) [Rivière et al. 1998]. Cette base de données permet à partager la connaissance entre les acteurs impliqués dans le processus de conception, et peut servir de référentiel de connaissance aux entrées ou à la sortie du processus de tâches. D'après Matta [Matta et al. 1999], la mémoire projet est une matérialisation possible de la mémoire d'entreprise : La mémoire de projet est une mémoire des connaissances et des informations acquises et produites au cours de la réalisation des projets.

III.3 Dynamique d'échange de connaissances au cours d'un processus de conception

L'évolution du projet illustre la nécessité de définir des règles strictes d'échange pour maintenir la concordance du système étudié. Ces règles concernent principalement l'échange des connaissances partagées au cours d'un processus de conception où les acteurs manoeuvrent des connaissances pour produire des concepts figure 2. Il est donc inévitable que les connaissances changeront dans le but de corriger des erreurs, d'adapter de nouvelle information, ou ajuster la représentation d'un domaine particulier. Cependant le mécanisme de versionnement peut être mené par des annotations et des validations des connaissances de la part d'autres acteurs impliqués dans le processus de conception.

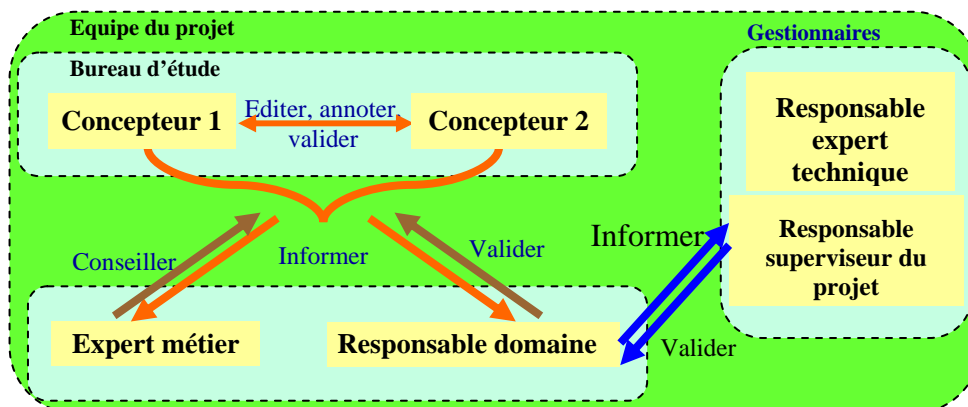


Figure 2. Echange de connaissances au sein d'une équipe d'un projet

Chaque acteur a une tâche spécifique dans un projet selon les engagements définis dans les contrats, le plan de gestion de qualité et les lois de domaine. La tâche de chaque acteur est employée pendant la planification du projet pour définir le rôle de l'acteur pendant les activités. Le modèle relationnel emploie les rôles des acteurs dans une activité pour déterminer les relations que les acteurs auront

avec les éléments de connaissances. Ces rôles peuvent être : auteur, conseiller, responsable, lecteur, etc.

Les acteurs projet participent dans la synchronisation des tâches en fonction des conditions d'exécution de ces derniers. De plus un acteur projet peut annoter le résultat d'un autre acteur à cause de la non concordance de ce résultat avec une exigence ou une contrainte projet. Ces annotations obligent le deuxième acteur à faire des modifications et corriger le paramètre en question. En cas de conflit les acteurs doivent se référer aux experts qui peuvent en plus annoter ou valider les résultats. La validation définitive des résultats est assurée par les gestionnaires de projet qui sont les premiers responsables de l'équipe du projet. Nous avons modélisé les différentes activités des concepteurs impliqués dans le processus de conception par un diagramme de classe figure 3. Ce diagramme représente la structure de notre modèle relationnel. Dans ce modèle nous identifions les types suivants de l'information :

- Le projet, est formé par des tâches,
- Les tâches, où les acteurs manipulent et produisent des connaissances,
- Les participants (acteurs) qui pourraient être une personne simple ou un groupe de personnes
- Les rôles qu'a chaque acteur dans une activité,
- Les relations entre les acteurs et les connaissances qui déterminent le droit d'accès sur des versions de connaissance,
- Les versions de connaissance qui peuvent être une simple instance de paramètre ou un groupe de version d'instance de paramètres.

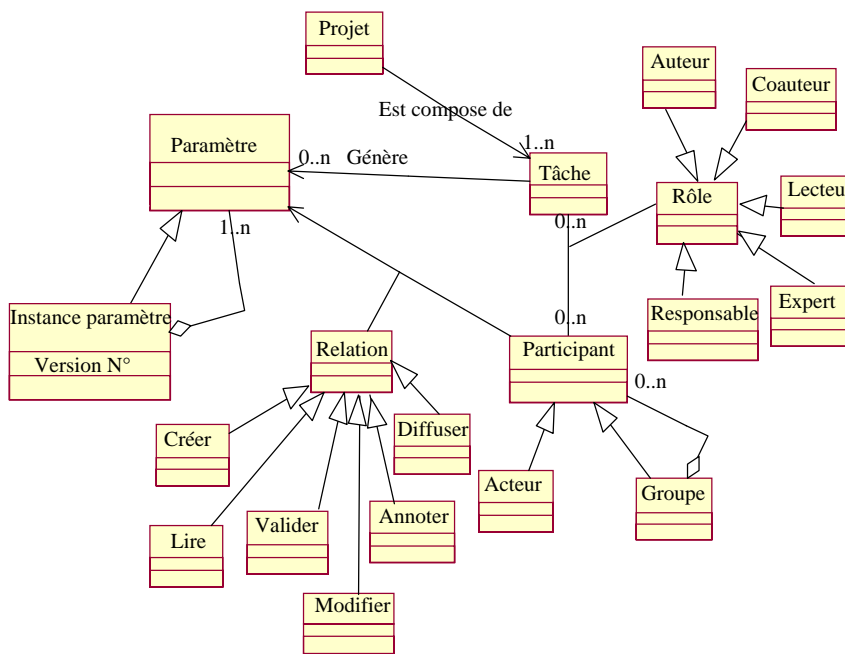


Figure 3. Modèle relationnel

IV Mécanisme de versionnement de connaissances

IV.1 La gestion du versionnement

Une question importante dans des activités de collaboration est comment commander des versions d'information. Maintenir des versions d'information sert trois objectifs :

- Enregistrer l'historique de l'information afin d'éviter la répétition des opérations ;

- Permettre des changements aux données sans références compromettantes aux versions précédentes de ces données
- Permettre l'inspection et la comparaison des versions.

La pratique actuelle des systèmes de gestion de documents fournit le contrôle de version, mais seulement au niveau du document. Pour la conception collaborative, le contrôle des versions est exigé à un niveau de détail plus fin pour une combinaison des raisons. Le nombre de personnes travaillant avec des données de conception est grand, l'ensemble des données de conception est grand, les documents ne servent pas toujours de base au stockage, il y a un fort rapport entre certaines données avec des documents [McGuinness et al. 2000]. Dans la littérature il y a plusieurs travaux qui ont abordé le versionnement, notamment le versionnement d'objets ([Kim et al. 1988], [Talens et al. 1993], [Van Leeuwen et al. 2003]), d'ontologie ([Klein et al. 2002], [Noy et al. 2003], [Compatangelo et al. 2004]) et de schémas de base de données ([Banerjee et al. 1987] [Lerner et al. 1990] [Ferrandina et al. 1995]).

IV.2 Naissances des versions de connaissances.

Pour émerger la nécessité du versionnement des connaissances nous avons implémenté les opérateurs de la théorie C-K [Hatchuel, et al. 2002], qui stipule que la conception oscille d'un espace à l'autre en co-construisant le produit et les connaissances qui le supportent, sur nos modèles. L'opérateur « expansion par déduction ou expérimentation » permet à l'espace de connaissance d'avoir une auto expansion. Dans notre modèle l'expansion des connaissances par déduction ou par expérimentation se traduit par l'instanciation des paramètres sur un projet donné. Cette instanciation permet à l'espace de connaissance d'avoir une auto expansion. Nous pouvons aussi considérer que les traces des opérations de versionnement des connaissances au cours du processus de conception permettent aussi d'avoir une auto expansion de l'espace de connaissance autrement dit du référentiel de connaissance processus de conception.

Au fur et à mesure de l'avancement du projet la base de donnée des valeurs des paramètres commence à changer. Au cours du processus de conception les modifications de valeurs de paramètres pour une tâche donnée engendrent des modifications sur les tâches qui soit elles partagent avec la première ses valeurs ou utilisent les paramètres de sortie de celle-ci comme paramètre d'entrée. D'où la nécessité de versionnement. Dans ce cadre chaque modification de valeur de paramètre donne une version de connaissance qui peut donner une version d'objet si elle a engendré une modification dimensionnelle, structurelle ou fonctionnelle sur le système à dimensionner. Toutes ces versions doivent avoir le label du projet lors de leur versionnement, car elles sont en réalité des versions d'instance de paramètre résultant d'une instance de tâche à ce projet.

IV.3 Versionnement des connaissances.

IV.3.1 Les versions d'instances de classes.

Dans notre modèle les connaissances versionnables sont généralement au niveau du projet, de ce fait le versionnement va apparaître au niveau des instance de classes du modèle de connaissances du produit et de son processus de conception. Une version d'instance est un état de la classe que l'utilisateur veut conserver. L'objectif des versions est de garder une trace dans le temps de l'évolution d'une instance. Le fait de créer une version d'une instance, lorsque celle-ci est modifiée, permet de garder les anciennes versions qui sont cohérentes par rapport aux autres instances. Les versions représentent un thème qui concerne plusieurs domaines tels que la CAO, les bases de données, les bases documentaires, etc. Les domaines techniques sont appropriés pour la gestion des versions tels que le domaine de la CAO. Cette approche engendre un nombre important de versions (versions d'instances) ce qui complexifie la navigation dans la bibliothèque des projet. Par conséquent il faut penser à un système de gestion des versions (création des versions) et de leur évolution

(propagation des versions) pour remédier à ce problème tout en préservant la cohérence des instances et la compatibilité des connaissances.

IV.3.2 Modèle de présentation des versions d'instance.

Versionner les modèles de connaissances, c à d la capacité de contrôler les modifications au niveau de ces modèles et les conséquences requises, c'est une fonctionnalité critique dans ce cadre qui traite la gestion des connaissances multiples. Le versionnement permet la description de l'évolution des connaissances par un ensemble de versions. Chaque version correspond à un état significatif de la connaissance. En fait, le versionnement nécessite une analyse complète des connaissances disponibles, qui est basée sur les changements présentés par la transformation d'une version de connaissance en une autre différente. Une telle analyse mène à la spécification de la traçabilité de l'évolution des connaissances qui permet de les capitaliser, en évitant d'éventuelles redondances, dans le but de les réutiliser sur d'autres projets. L'historique de la conception d'un produit, se traduit par un arbre de versions qui évolue dans le temps.

Au cours du processus de conception, des contraintes et des dépendances de partage [Djeraba, 1993] sont spécifiées entre les versions d'instances d'une même classe. Elles décrivent les attributs partagés, les relations de dérivation des valeurs d'attributs et les dépendances générales entre les attributs des versions. La modification d'un attribut d'une version peut provoquer la mise à jour des autres versions (par exemple, recopie de valeurs) pour maintenir les contraintes et les dépendances. Les versions sont liées par des liens de dérivation et forment un arbre de versions. Une dérivée représente une succession de changement d'état d'une version. Par contre, une alternative représente une évolution différente d'une même entité qui traduit le choix d'une nouvelle solution.

Les liens de dérivation entre les versions indiquent qu'une version est dérivée d'une autre existante : un nouvel état de connaissance est créé d'un autre état de la même entité de connaissance (Cv1 est dérivé de C). Les versions dérivées de la même version s'appellent les alternatives (versions d'une version) (Cv2.1 et Cv2.2 sont des alternatives dérivées de Cv2) (figure 4).

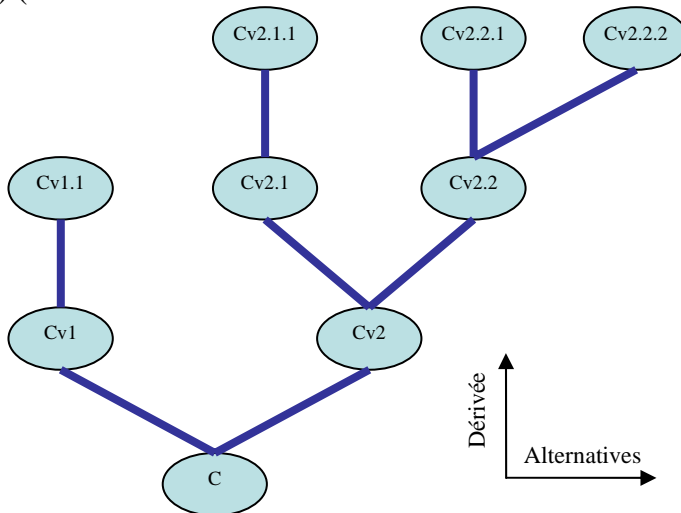


Figure 4 : Arbre de versions

IV.4 Classification des versions

Les versions de connaissances sont distinguées par la combinaison d'un nombre de versions majeures « M » et un nombre de versions mineures « m » sous la forme de « M.m ». La soumission d'une version au système peut mener à une nouvelle version majeure ou à une nouvelle version mineure :

- Versions majeures : il y eu une grande modifications au niveau des attributs de l'instance de la classe (versions originales ou mère). Elle est créée pour un nouveau type du concept ou à l'apparition d'une nouvelle technologie.

- Version mineures : créée à partir d'une version majeure, elle hérite les attributs de la version majeure mais avec des valeurs différentes.

Les versions mineures indiquent la dernière compatibilité, qui signifie que la version peut être employée au lieu des versions mineures précédentes de la même version majeure. Par exemple, ajouter une propriété mène à une nouvelle version mineure parce qu'elle ne transforme pas sa fonctionnalité dans les endroits où sans cette propriété il a été exécuté.

Les nouvelles versions majeures ne sont pas compatibles avec les derniers, ce qui signifie qu'elles ne peuvent pas être employées au lieu d'aucune version précédente. Les modifications telles que modifier des propriétés ou changer le type de propriétés mèneront généralement à de nouvelles versions majeures figure 5. Si une soumission de une nouvelle version majeure ou mineure, est déterminée en premier lieu par l'utilisateur, alors le système imposera des versions majeures quand il détecte l'incompatibilité avec l'antérieur. L'apparition d'une version mineure plus récente de son type est généralement possible et peut être faite probablement automatiquement. Un nombre incrémenté de révision est assigné chaque fois qu'une révision est commise ou une version est soumise [Van Leeuwen et al. 2003].

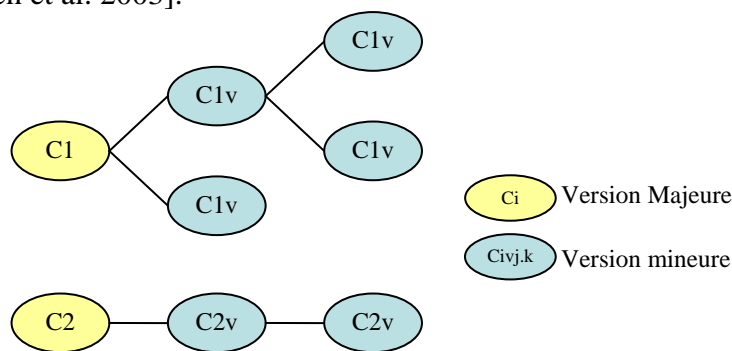


Figure 5 : Exemple de propagation de versions majeures et de versions mineures

IV.5 Agrégation des connaissances.

Notre démarche consiste à créer des versions de connaissances pour faciliter leur réutilisation. Les versions peuvent être créées soit en faisant des modifications sur les connaissances relatives au processus de conception (nouvelle méthode de traitement) ou sur les connaissances du produit (nouvelle technologie). Les révisions peuvent être créées au cours de l'exécution du processus de conception à travers les annotations et les vérifications entre acteurs. Un troisième mode de création de version sera en agrégeant des connaissances au près d'autres processus de conception ressemblant à celui en cours d'étude.

IV.5.1 Principe de l'agrégation des connaissances

La figure 6 montre la possibilité d'agréger des connaissances à partir d'anciens projets dont les représentations de produit et les processus de conception sont stockés sous forme de versions de connaissances dans la bibliothèque des projets (ils portent le label du projet et un numéro de version). L'agrégation des connaissances consiste d'une part à emprunter certains éléments de connaissances relatives aux produits déjà conçus et les intégrer dans les connaissances du produit en cours de conception. D'autre part il serait bénéfique, en terme de temps et de coût de la conception, d'agréger des parties des processus de conception déjà exécutés (tâches élémentaire de calcul sous forme de légende .xls, des ressources, etc.) et les intégrer entre les tâches d'exécution d'un nouveau processus de conception.

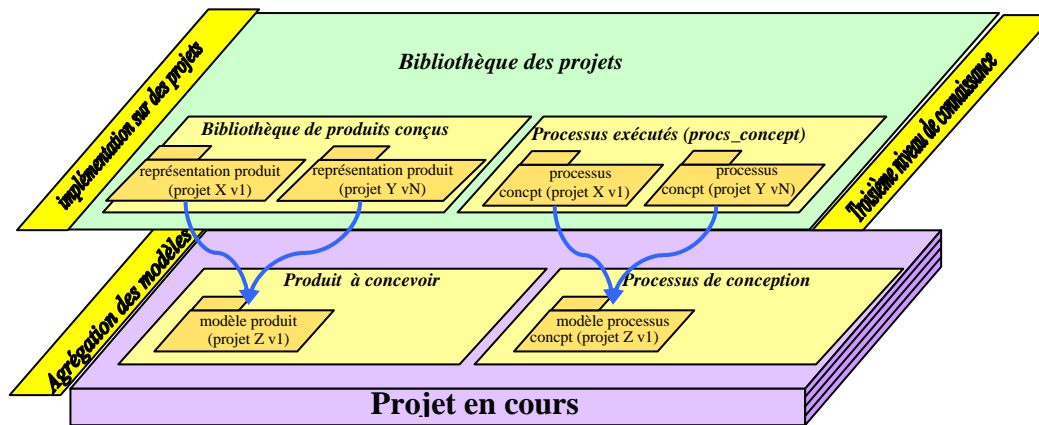


Figure 6 : Agrégation d'anciens éléments de connaissances à nouveau projet

IV.5.2 Evolution du mécanisme d'agrégation des connaissances

La version issue d'une agrégation d'éléments de connaissances sera classifiée comme étant une version majeure, qui à son tour peut être versionnée en cas de nécessité et ils seront enregistrées comme étant des versions mineures. La figure 7 illustre un exemple d'évolution d'un mécanisme de versionnement et d'agrégation de connaissances. La version majeure C1 a évolué par l'apparition des versions mineures C1.1 et C1.2 qui à leurs tours ont évoluées par la naissance d'autres versions mineures respectivement C1.1.1 et C1.2.1 et C1.2.2 etc. l'agréations de certains éléments de connaissances des versions mineures C1.1.1 et C1.2.2 à un nouveau concept engendre l'apparition d'une nouvelle version majeure C2. Cette dernière peut évoluer normalement comme tout autre version majeure à travers l'apparition de versions mineures portant son label tel que C2.1.

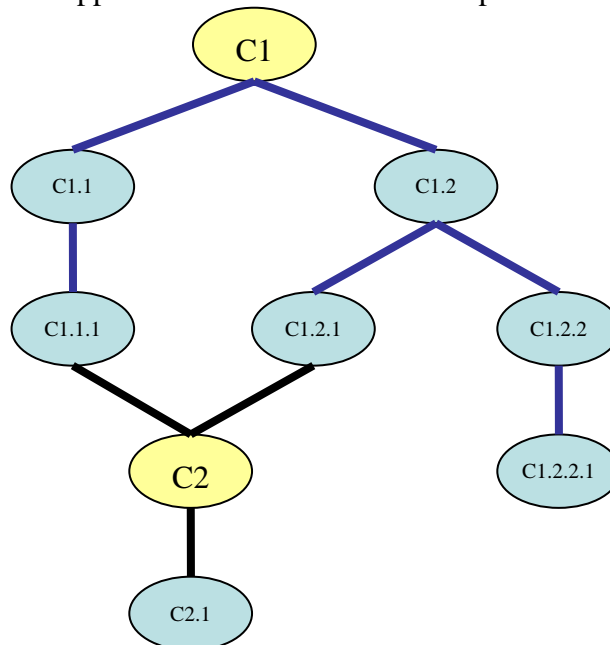


Figure 7 : Evolution des versions issues d'une agrégation de connaissances

IV.6 Versions des processus de conception

Pour exécuter une tâche, le concepteur peut disposer de plusieurs processus de conception (calcul, choix, etc.) selon certain critères de base tels que les hypothèses de simplification, les différentes normes, l'apparition de nouvelles technologie, les lois en vigueur, etc. Cette diversité de méthode de calcul mène au versionnement des processus de conception. C'est à l'expert de domaine de choisir la version de processus adéquate à imposer aux acteurs de projet en fonction des exigences du projet.

Les instances de tâches (version de processus de conception) sont associées aux instances de paramètres (versions de connaissances). En effet, si une tâche doit être refaite, obligatoirement, ses sorties (paramètres) vont être modifiés. Donc la création d'une nouvelle instance de tâche crée une nouvelle instance de paramètres de sortie de la tâche. La modification d'une valeur de paramètre sur un projet donné implique que les tâches dont ce paramètre est en entrée soient à redéroulées. Donc la création d'une nouvelle instance de paramètre implique la création d'une nouvelle instance de toutes les tâches dont ce paramètre est en entrée.

IV.7 Evolution du mécanisme de versionnement de connaissance

IV.7.1 Dynamique des connaissances au cours d'un processus de conception.

Au cours du processus de conception les connaissances sont instamment modifiées par les acteurs qui les manipulent, particulièrement les paramètres. Nous avons classé les paramètres d'entrée en fonction de leur apparence sur les feuilles d'exécution de tâches :

- Entée externe propre: si sur la ligne correspondante à ce paramètre on trouve qu'il sert d'entrée que pour une seule tâche.
- Entrée externe partagée : si sur la ligne correspondante à ce paramètre on trouve qu'il sert d'entrée externe pour plusieurs tâches.
- Entrée interne propre : si sur la ligne correspondante à ce paramètre on trouve qu'il est un paramètre de sortie d'une tâche et peut servir d'entrée pour une autre tâche.
- Entrée interne partagée : si sur la ligne correspondante à ce paramètre on trouve qu'il est un paramètre de sortie d'une tâche et peut servir d'entrée pour d'autres tâches.

En utilisant une matrice d'antériorité nous allons pouvoir classer tous les tâches. Sur le tableau utilisé pour la classification des tâches savoir à quelle tâche appartient chaque paramètre de sortie et quels sont les tâches qui l'utilisent comme paramètre d'entrée. Le responsable doit cocher sur la matrice d'antériorité les cases correspondantes aux tâches utilisant comme paramètre d'entrée la sortie de la tâche étudié. Cette démarche nous permet de tracer le workflow du processus de conception et définir les conditions d'ordonnancement de chaque tâche. Exemple : Soit un processus de conception formé par sept tâches réalisées par différents acteurs travaillant en collaboration figure 8. Un graphe d'antériorité nous permet de définir le workflow du processus en fonction des paramètres que partagent les concepteurs.

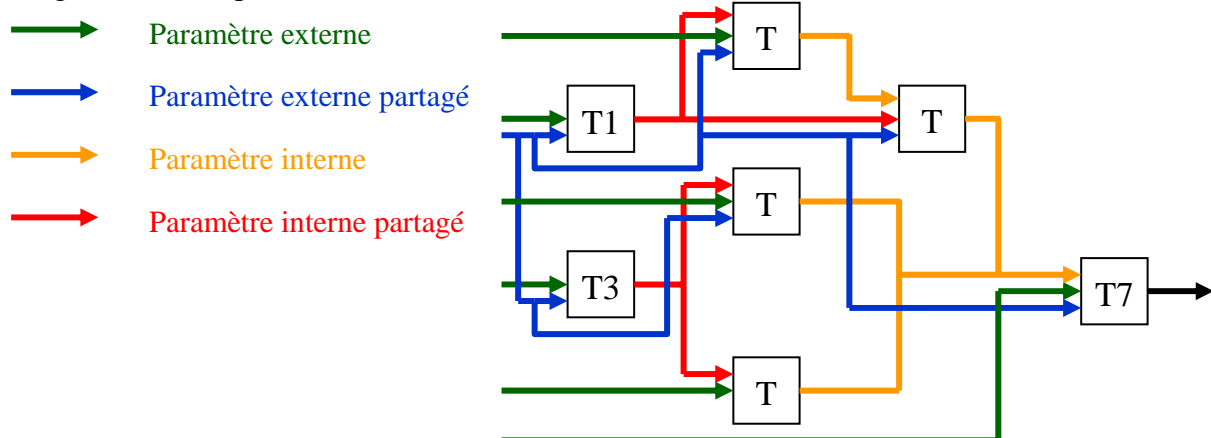


Figure 8 : Exemple de succession de tâches définissant un processus de conception

IV.7.2 Gestion des versions de connaissances

Une question importante dans des activités de collaboration est comment contrôler des versions d'information. Maintenir des versions d'information sert trois objectifs : pour enregistrer l'historique de l'information afin d'éviter la répétition des opérations ; pour permettre des changements aux données sans références compromettantes aux versions précédentes de ces données et pour permettre l'inspection et la comparaison des versions.

La pratique actuelle des systèmes de gestion de document fournit le contrôle de version, mais seulement au niveau de document. Pour la conception collaborative, le contrôle des versions est exigé à un niveau de détail plus fin pour une combinaison des raisons. Le nombre de personnes travaillant avec des données de conception est grand, l'ensemble des données de conception est important, les documents ne servent pas toujours de base au stockage, il y a un fort rapport entre certaines données avec des documents [McGuinness et al. 2000].

Le cadre de FBM (Feature Based Modelling) a un fort soutien pour le contrôle des versions à la fois pour le type de caractéristiques et leur instance. L'édition des caractéristiques de données (des types et des instances) aura lieu par l'intermédiaire d'un mécanisme de contrôle et d'exécution, à travers lequel les utilisateurs obtiennent le privilège d'édition provisoires. Quand les données sont vérifiées pour l'édition, les versions précédentes peuvent continuer à être employées. Après l'édition, les données peuvent être soumises comme nouvelle version, ou être employées comme révision (version temporaire). Les révisions des caractéristiques de données sont inférieures aux versions dans le sens qu'elles ne peuvent pas encore être activement employées dans une opération de modélisation, mais seulement pour une future édition (modification) de données. Ceci réduit le nombre de versions et permet la distinction entre ceux dont la soumission a un vrai intérêt et ceux ayant seulement un statut intermédiaire. Exemple, si un acteur crée une des connaissances en exécutant la tâche de conception qui lui est attribuée. Ces connaissances seront enregistrées temporairement tout au long du processus de conception comme étant une révision susceptible d'être modifiée dans le temps, de ce fait nous obtenons une nouvelle révision après chaque modification figure 9.

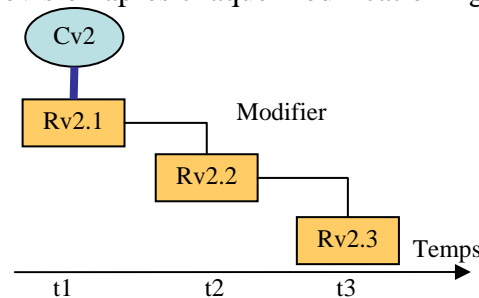


Figure 9 : Création dynamique de révisions à partir d'une version de connaissances

Le domaine des connaissances est continuellement créé et toujours modifié par les acteurs. La gestion efficace de cette connaissance a besoin ainsi de sa vérification, annotation et validation continues (V&V). Dans ce cadre les révisions créées au cours de l'évolution dynamique du processus de conception à travers les modifications annotations seront validées à la fin du processus, à ce moment la révision sera définitive et elle se transforme en une version susceptible d'être enregistrée dans la bibliothèque de projet.

Si un acteur projet crée une révision de connaissance (paramètre) au cours d'un processus de conception elle serait enregistré dans une base de donnée temporaire en attendant la réaction des acteurs qui l'emploi dans leurs tâches. Ces derniers peuvent annoter cette révision en fonction des exigences de leurs tâches, dans ce cas l'auteur de la révision de connaissance doit la modifier en tenant compte des annotations de son collègue. En cas de conflit c'est au responsable de domaine d'intervenir pour annoter ou valider cette révision de connaissance. La révision de connaissance ne peut être enregistré dans la bibliothèque de projet comme étant une version de connaissance sauf si

tous les acteurs qui la partagent ainsi que le responsable de domaine l'ont validé. La figure 10 donne une idée sur ce mécanisme.

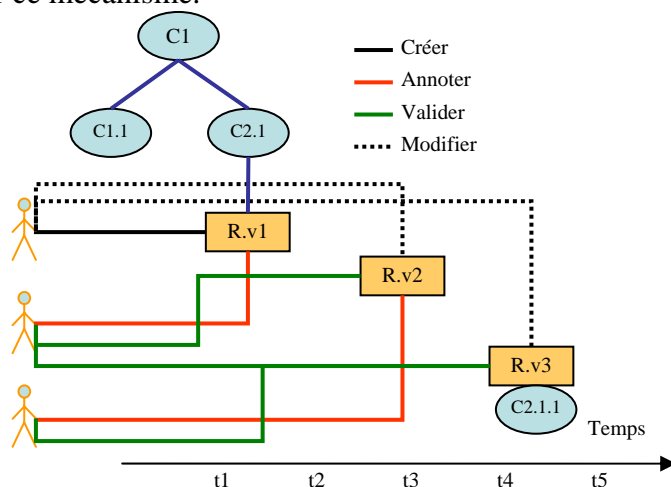


Figure 10 : Exemple de mécanisme de transformation d'une révision en version de connaissance

V Application sur les systèmes de climatisation automobile

V.1 Versionnement des processus de conception

Le processus de conception des systèmes de climatisation pour voiture automobile (figure 8) est trop détaillé, d'autres processus peuvent être utilisés si le constructeur décide de diminuer le coût de la conception de cet organe. Ces processus ont généralement la même structure que le précédent mais ils négligent certains apports calorifiques (externes ou internes) n'ayant pas trop d'influence sur la valeur de la puissance calorifique du compresseur à choisir. Les simplifications peuvent aller jusqu'au dimensionnement de la clim en fonction du volume du véhicule à conditionné, en multipliant ce paramètre par un facteur pour avoir la puissance frigorifique à installé.

La figure 11 présente un exemple de versionnement des paramètres relatifs au dimensionnement d'un système de climatisation de voiture automobile en fonction du volume de l'habitacle et en fonction de la cylindrée du moteur.

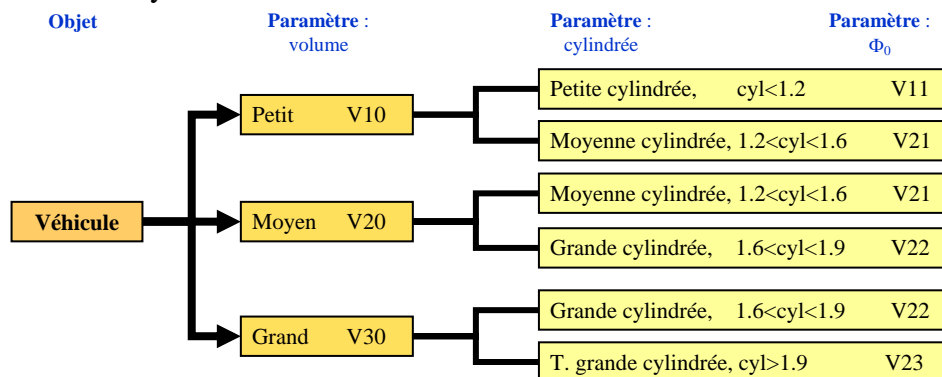


Figure 11 : Versionnement en fonction de paramètres déterminants

La figure 12 présente un exemple de versionnement des paramètres des installations de climatisation de voitures automobiles en fonction du choix du fluide frigorigène (interdiction des fluides chlodofluoré (R22) et apparition de nouvelles techniques (CO2)) et de la technologie relative à la source de puissance du compresseur (compresseur attelé ou motocompresseur) qui est fonction de la puissance effective du moteur de la voiture automobile.

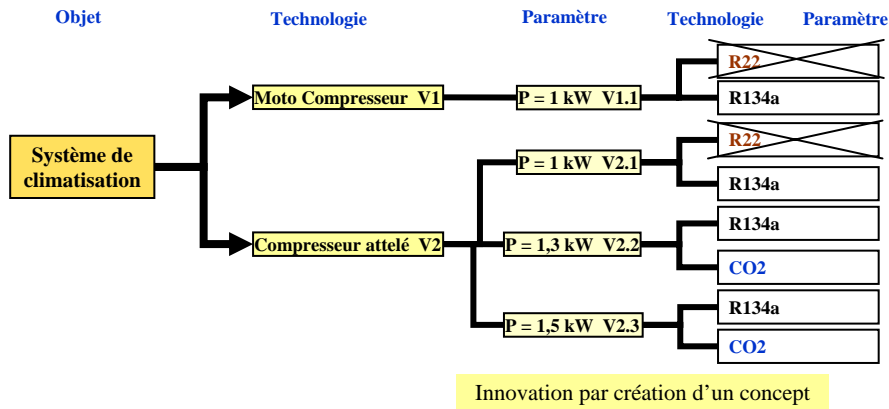


Figure 12 : Versionnement suite à une innovation ou une loi d'interdiction

V.2 Versions de connaissances

Pour améliorer la réutilisation des connaissances capitalisées et les contrôler facilement, ils doivent être stockés comme étant des versions de connaissance. Les traces des opérations de versionnement de connaissance pendant le processus de conception laissent avoir également une auto expansion de l'espace de connaissance. En d'autres termes, c'est une auto expansion du référentiel de processus de conception.

La figure 13 illustre un exemple d'instance du paramètre diamètre d'aspiration (versions de connaissances) sur différents projets. En outre elle montre le versionnement de la valeur d'un paramètre à un projet particulier. Ce modèle correspond à un versionnement de connaissances.

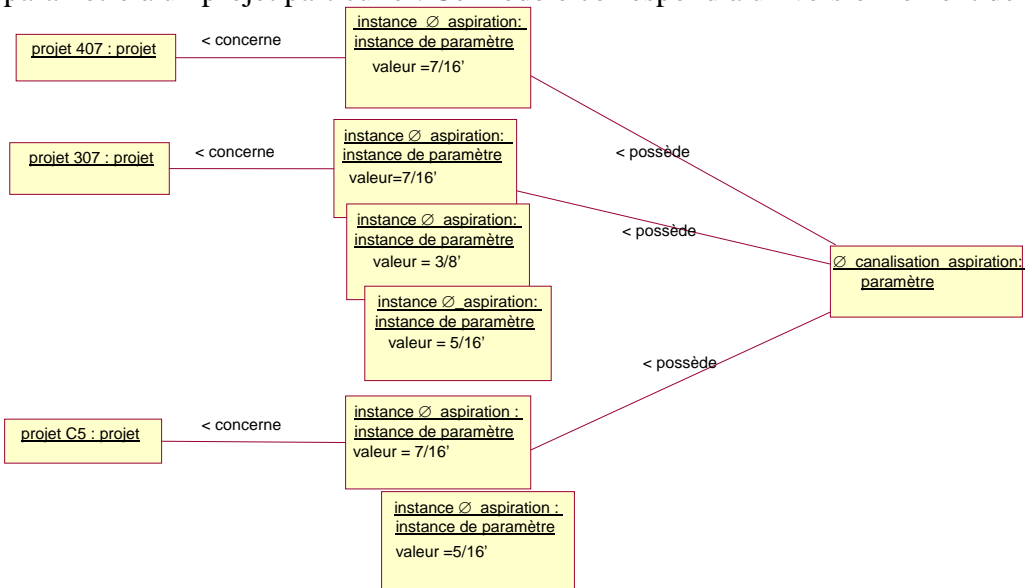


Figure 13 : Exemple d'instance du paramètre sur différents projets

Au cours du processus de conception, le versionnement des connaissances est focalisé sur des versions de valeurs de paramètres issues d'un même processus de conception. Le modèle conceptuel, figure 7-30, montre que chaque paramètre peut avoir des valeurs lorsqu'il est instancié à un projet donné. Les valeurs de paramètres peuvent être versionnées au cours de l'exécution d'un processus de conception collaboratif, en fonction des modifications qu'elles peuvent avoir lors de leurs échanges entre les différents acteurs intéressés par ce paramètre. Les versions doivent porter le label du projet car elles sont des versions d'instance de paramètre résultant d'une instance de tâche appartenant au processus de conception de ce projet. Elles seront sauvegardées dans une base de données propre au projet (référentiel de connaissances de projet), pour tracer l'historique de son exécution. L'ensemble des référentiels de projets constitue la bibliothèque de projets.

V.3 Versionnement et agrégation des connaissances

La figure 14 présente un scénario de versionnement et d'agrégation de connaissances relatif à un processus de conception de systèmes de climatisation automobile. Le projet 307 par exemple contient différentes plateformes (versions d'objet) (petite, petite avec vitres teintées, grande et grande avec vitres teintées). Si pour la version d'objet petite nous avons utilisé la version de connaissances V1 (version paramètre et processus de conception 'légende xls'). Cette dernière a évolué par la création des versions V1.1 et V1.2 relatives respectivement aux versions d'objet petite et vitre teint et grande. V1.2 à son tour a évolué au niveau du projet 407 par la création de la version V1.2.1.

La conception du système de climatisation relatif à la version d'objet du projet 307 grande et vitre teint, nécessite la réconciliation de certains éléments de connaissances à partir de V1.1 et V1.2. Ce mécanisme engendre l'apparition d'une nouvelle version V2 par agrégation de connaissance.

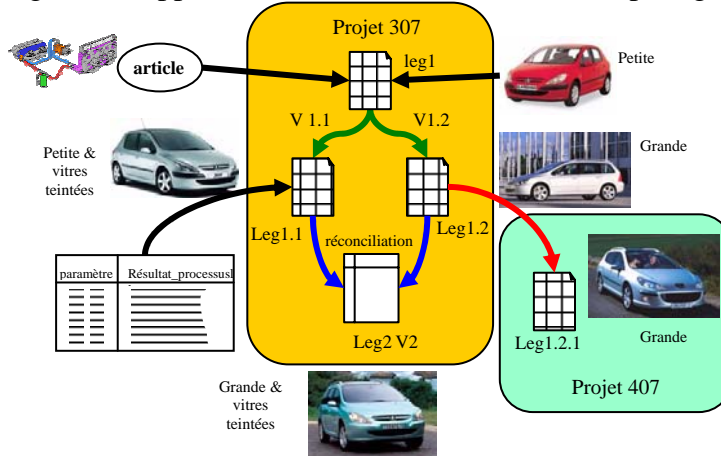


Figure 14 : Exemple de versionnement et agrégation des connaissances

VI DIEKOVER un outil de conception par le versionnement de connaissance

L'étude a conduit au développement d'une maquette informatique appelé «DISKOVER» (DesIgn with KnOwledge VERsioning) figure 15. DISKOVER propose une structure permettant de recueillir les connaissances relatives à la conception multi acteurs ainsi qu'au produit associé. La maquette numérique de DISKOVER est conçue sur un environnement WinDev.



Figure 15 : Page d'accueil

Pour la consultation des connaissances, le logiciel permettra à un acteur de :

- Consulter les tâches du processus et leur état : Un acteur pourra consulter les tâches du processus de conception qui lui concerne (données d'entrée, données de sortie, connaissances associées....) ainsi que leurs états (prête, non faisable...).
- Consulter les paramètres du produit : Il pourra accéder aux paramètres dont il est responsable et aux paramètres qui lui sont nécessaires pour réaliser ses tâches ainsi que leurs résultats. Pour l'opérationnalisation des connaissances (dynamisation de connaissances), les méthodes traditionnelles obligeaient les concepteurs à se reporter à des guides de conception, notes internes, livres de

connaissances, intranet pour les aider à réaliser une tâche et à prendre en compte le retour d'expériences des projets passés. Ici, on propose d'associer à chaque tâche les connaissances nécessaires à sa réalisation et accessibles par simple clic.

- Assister les acteurs dans le déroulement du processus : DISCOVER distribuera aux acteurs concernés ses tâches ainsi que leurs connaissances (avec contraintes et retour d'expériences) associées selon l'ordre chronologique établis par le déroulement du processus de conception. Il préviendra ainsi l'utilisateur concerné qu'il a une tâche à réaliser (tâche prête), l'acteur ouvre cette tâche, la réalise et valide ou lance son exécution si celle ci est réalisée par un logiciel externe (calcul sous EXCEL par exemple) pour enfin lire le résultat et valider.

Pas de la transmission	LARGEUR STANDARD							
	6.35	9.52	12.7	19.05	25.4	38.1	50.8	76.2
1.4"	3/8"	1/2"	3/4"	1"	1 1/2"	2"	3"	
XL (5.08 mm - 1.5")	0.25	0.37						
L (9.52 mm - 3/8")			0.50	0.75	1.00			
H (12.7 mm - 1/2")				0.75	1.00	1.50	2.00	3.00
Coefficient F2	0.15	0.28	0.42	0.71	1.00	1.56	2.14	3.36

Figure 16. Fiche d'exécution de tâche

Avant d'accéder au processus de conception, l'acteur doit s'identifier dans le système en tant qu'utilisateur par un login et un mot de passe. Une fois identifié, l'acteur peut consulter les tâches qu'il doit sur lesquelles il va travailler. Il doit choisir celles dont les conditions d'exécution et d'ordonnement sont prêtes, à ce moment le bouton exécuter sera activé.

En cliquant sur le bouton exécuter, la fiche de tâche apparaît. Les paramètres internes figurant la fiche ne peuvent pas être modifiés, mais annotés. L'acteur aura la possibilité de saisir les valeurs des paramètres externes et de consulter les ressources de la tâche (tableaux, courbes...) voir figure 16. Une fois la fiche est remplie, l'acteur clique sur le bouton exécuter pour lancer le processus de calcul. Par la suite l'acteur peut consulter les résultats et les enregistrer avant de fermer la fiche.

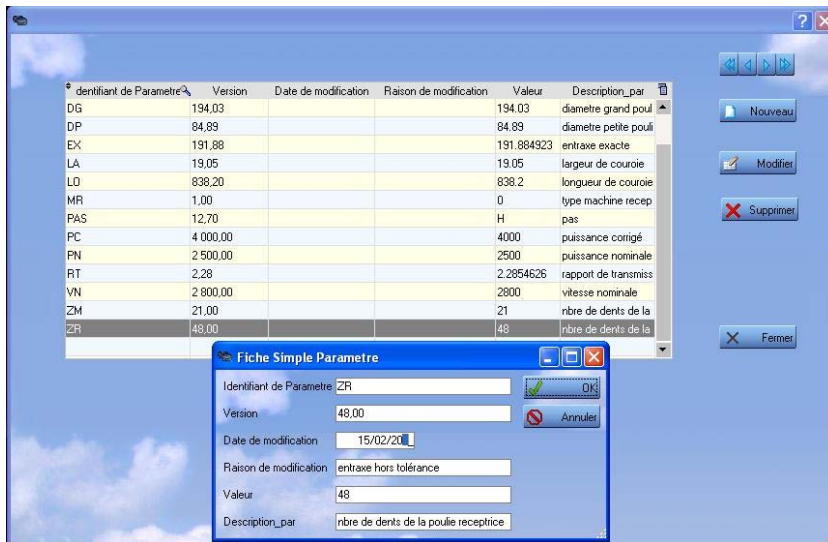
En cliquant sur enregistrer, le concepteur donne l'ordre au système d'enregistrer les résultats comme étant une révision et d'envoyer les valeurs des paramètres de sortie aux acteurs dont ils ont besoin. Si les valeurs sont déjà stockées dans la base, le logiciel envoie un message à l'acteur concerné lui indiquant que ces valeurs existent avec une version bien déterminée figure 17.

Parametres:	Version précédente:	Version actuelle:
Coefficient de service:	2.50	2.50
Vitesse nominale:	3 000.00	3 000.00
Puissance nominale:	2 000.00	2 000.00
Pas:		12.70
Rapport de transmission:		3.00
Nbre de dents de la poulie motrice:		20.00
Nbre de dents de la poulie receptrice:	60.00	60.00
Diametre grande poulie:	242.45	242.45
Diametre petite poulie:	80.85	80.85
Longueur de courroie:	914.40	914.40
Entraxe:	185.27	185.27
Largeur standard:	19.05	19.05
Code courroie:	75.00	75.00

Figure 17. Etat de version des paramètres

Le responsable métier doit s'identifier dans le système en tant qu'administrateur par un login et un mot de passe pour accéder au menu. Le menu permet à l'administrateur de rentrer pour consulter les différents :

- Projets : consulter et ajouter des projets
- Acteurs : consulter, ajouter et supprimer les acteurs d'un projet. L'administrateur peut aussi affecter les acteurs à des tâches.
- Paramètres : consulter, ajouter, modifier ou supprimer les paramètres relatifs à un processus (figure 18).



Identifiant de Parametre	Version	Date de modification	Raison de modification	Valeur	Description_par
DG	194.03			194.03	diametre grand poul
DP	84.89			84.89	diametre petite poulie
EX	191.88			191.884923	entree exacte
LA	19.05			19.05	largeur de courroie
LD	838.20			838.2	longueur de courroie
MR	1.00			0	type machine recep
PAS	12.70			H	pas
PC	4 000.00			4000	puissance corrigé
PN	2 500.00			2500	puissance nominale
RT	2.28			2.2854626	rapport de transmiss
VN	2 800.00			2800	vitesse nominale
ZM	21.00			21	nbre de dents de la
ZR	48.00			48	nbre de dents de la

Figure 18. Table des paramètres.

- Processus : attribuer des processus d'exécution pour une tâche donnée. Les processus sont des feuilles de calcul sur Excel pouvant à leur tour être versionnés en fonction de la nécessité.
- Tâches : consulter l'ensemble des tâches et leurs instances du processus de conception sur le projet choisi. Il a en outre la possibilité d'ajouter, modifier ou supprimer des tâches.
- Ressources : consulter, ajouter, modifier ou supprimer les ressources relatives à une tâche.

VII Conclusions et perspectives

Le versionnement de connaissance pendant le processus de conception nous permettent d'avoir une auto expansion de l'espace des connaissances, en d'autres termes le référentiel de connaissance du processus de conception. Nous avons proposé dans ce cadre un mécanisme de versionnement de connaissances basé sur les instance de classe (objet ou processus de conception). Ce mécanisme engendre un système de gestion des versions (création des versions) et de leur évolution (propagation des versions) tout en préservant la cohérence des instances et la compatibilité des connaissances. Les versions peuvent être créé soit à travers des modifications sur les connaissances relatives au processus de conception ou sur les connaissances du produit. Un troisième mode de création de version sera en agrégeant des connaissances au près d'autres processus de conception (réutilisation des connaissances). Nous avons distingué deux formes de versions, majeure et mineure. Les versions majeures ne sont pas compatibles avec les versions précédentes (pas d'héritage d'attributs), elle représente en fait une version mère. Les modifications des propriétés mèneront à de nouvelles versions majeures. Les versions mineures sont compatibles avec les versions précédentes, ils dérivent soit d'une version majeure ou d'une versions mineures précédentes déclinant d'une même version majeure.

Au cours d'un processus de conception, si un acteur crée des connaissances en exécutent sa tâche de conception, elles seront enregistrées temporairement tout au long du processus de conception comme étant une révision susceptible d'être modifié. les révisions créés au cours de l'évolution dynamique du processus de conception à travers les modifications annotations seront validées à la fin du processus, à ce moment la révision sera définitive et elle se transforme en une version susceptible d'être enregistrée dans la bibliothèque de projet. Pour valider ce mécanisme nous avons proposé un scénario de l'évolution dynamique du processus de conception à travers les relations interactives entre les acteurs impliqués dans le processus de conception et le versionnement des connaissances. Pour valider ces concepts, ils ont été appliqués sur la conception fonctionnelle des systèmes de climatisation automobile.

En perspectives nous souhaitons étudier la pertinence du versionnement sur les processus de fabrication (conception intégrée).

Principales références bibliographiques :

- **[Banerjee et al. 1987]** : Banerjee J., Kim W., Kim H. J., and Korth H. F., Semantics and Implementation of Schema Evolution in Object-Oriented Databases. In: Proceedings of the ACM-SIGMOD Annual Conference, May 1987, pp 311-322.
- **[Compatangelo et al. 2004]** : Compatangelo E., Vasconcelos W. et Scharlau B., Managing ontology versions with a distributed blackboard architecture IA 2004.
- **[Djeraba, 1993]** : Djeraba C. Composite objects and dependency relationship in engineering. In Proceeding AIENG, Toulouse 1993.
- **[Dupinet, 1991]** : Dupinet E., Contribution à l'étude d'un système informatique d'aide à la conception de produits mécaniques par la prise en compte des relations fonctionnelles, Thèse Ecole Centrale de Paris, 1991.
- **[Eckert et al. 2003]** : Eckert C., Boujut J-F, "The role of objects in design co-operation: communication through physical or virtual objects", Computer Supported Collaborative Work, vol. 12, pp 145-151, 2003.
- **[Harani, 1997]** : Harani Y., Une approche multi-modèles pour la capitalisation des connaissances dans le domaine de la conception, Thèse Institut National Polytechnique Grenoble, 1997.
- **[Hatchuel et al. 2002]** : Hatchuel A., and Weil B., "C-K Theory: Notions and Applications of a Unified Design Theory", Proceedings of the Herbert Simon International Conference on " Design Sciences ", Lyon, 2002, pp. 22.
- **[Kim et al. 1988]** : Kim W., Chou H.T., "Versions of schema for object oriented databases", Proceedings of the 14th VLDB Conference, Los Angeles, California, 1988. pp. 148-159.
- **[klein et al. 2002]** : Klein M., Kiryakov A., Ognyanov D., et Fensel D., Ontology versioning and change detection on the web. In 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW02), Spain, 2002. pp 197-212
- **[Lerner et al. 1990]** : Lerner B.S., Habermann A.N., Beyond schema evolution to database reorganisation. In Proc. ECOOP '90, Octobre 1990.
- **[Mahe, 1998]** : Mahe H. Une démarche et des outils d'aide à la conception d'ensembles mécaniques dans l'industrie automobile, dossier - Signaux n°92 - Mars 1998.
- **[Matta et al. 1999]** : Matta N., Corby O. et Ribière M. ; Définition d'un modèle de mémoire de projet, Rapport de recherche INRIA N° 3720 Juin 1999.
- **[McGuinness et al. 2000]** : McGuinness D. L., Fikes R., Rice J. et Wilder S., « An environment for merging and testing large ontologies. », In A. G. Cohn, Proceedings of the Seventh International Conference (KR2000). Morgan Kaufmann Publishers, San Francisco, CA, 2000.
- **[Menand 2002]** : Menand S., « Modélisation pour la réutilisation du processus de conception Multi acteurs de produits industriels » Thèse de Doctorat de l'institut National Polytechnique – Grenoble, 2002.
- **[Noy et al. 2003]** : Noy N. F. et Musen M. A. Ontology Versioning as an Element of an Ontology-Management Framework in Stanford Medical Informatics SMI proceeding March 31, 2003 at Stanford, CA 94305, USA.
- **[Ouazzani, 1999]** : Ouazzani A., Représentation dynamique du processus de conception : une perspective de capitalisation de l'historique de conception, Thèse Ecole Centrale de Paris, 1999.
- **[Ribière et al. 1998]** : Ribière M., Matta N. et Cointe C. "A proposition for managing project memory in concurrent engineering", Proceedings of International Conference on Computational Intelligence And Multimedia Applications (ICCIMA'98), Churchill, Australia, 1998.
- **[Saucier, 1997]** : Saucier A., Un modèle multi-vues du produit pour le développement et l'utilisation de systèmes d'aide à la conception en ingénierie mécanique, Thèse Ecole Normale Supérieure de Cachan, 1997.
- **[Sellini, 1999]** : Sellini F., « Contribution à la représentation et à la vérification des modèles de connaissances en ingénierie d'ensembles mécaniques », Thèse Ecole Centrale de Paris, 1999.
- **[Talens et al. 1993]** : Talens G., Oussalah C. et Colinas M. F. Versions of simple and composite objects. In Proceedings of the 19th conference on Very Large Data Bases, Dublin, Ireland, August 24-27, 1993.
- **[Tichkiewitch, 1996]** : S. Tichkiewitch, Specifications on integrated design methodology using a multi-view product model, Engineering System Design and Analysis Conference – ASME'96, Montpellier, 1996.
- **[Tollenaere, 1994]** : Tollenaere M., Contribution à la modélisation de connaissances pour la conception mécanique, Habilitation à diriger des recherches, Université Joseph Fourier Grenoble I, 1994.
- **[Van Leeuwen et al. 2003]** : Van Leeuwen J.P. et Fridqvist S. "Object Version Control for Collaborative Design", Proceedings of the 9th EuroIA International Conference Istanbul, TR, October 8–10, 2003, pp. 129-139.
- **[Vargas, 1995]** : Vargas C., Modélisation du processus de conception en ingénierie des systèmes mécaniques. Application à la conception d'une culasse automobile. Thèse de l'ENS Cachan, 1995.

- **[Wallace et al. 1997]** : Wallace K., Matheson J., Hogue C. et Isgrove D., Three Years of Running an Integrated Design Project at Cambridge, Proceedings of ICED, Tampere, August 1997.
- **[Yvars, 2001]** : Yvars P.A., Contribution à la représentation des connaissances en ingénierie intégrée de produits et de systèmes automatisés de production, Habilitation à diriger des recherches, Institut National Polytechnique de Grenoble, 2001.
- **[Zouari et al. 2005]** : Zouari A., Tollenaere M., Maalej A. et Ben Bacha H. "Assistance à la conception collaborative par la capitalisation et la réutilisation des connaissances", International conference: design and mechanical system modelling, CMSM'2005 Hammamet, Tunisia, March 23-25, 2005.
- **[Zouari et al. 2007]** : Zouari A., Tollenaere M., Ben Bacha H. et Maalej A. " Can C-K theory apply to KBE (knowledge based engineering) Systems?", to appear in International Journal of Product Lifecycle Management IJPLM, Inderscience publisher 2007.

Listes de publications de l'auteur :

- **Nom de publication :**

Application of a multi-actors design model to the car air-conditioning system functional design : colloque IEEE-Smc du 6 au 9 octobre 2002 à Hammamet-Tunisie : Alaeddine ZOUARI, Michel TOLLENAERE et Sébastien MENAND

Capitalisation des connaissances de domaine à travers un référentiel métier : colloque IPI 2004 les 22 et 23 janvier 2004 à autrans-France : Alaeddine ZOUARI, Michel TOLLENAERE, Habib BEN BACHA et Aref MAALEJ.

Assistance à la conception collaborative par la capitalisation et la réutilisation des connaissances : Congrès International CMSM'2005 du 23 au 25 mars 2005 Hammamet-Tunisie : Alaeddine ZOUARI, Habib BEN BACHA, Aref MAALEJ, Michel TOLLENAERE.

Can C-K theory apply to KBE (knowledge based engineering) Systems? : Accepté au IJPLM International Journal of Product Life cycle Management : Alaeddine ZOUARI, Habib BEN BACHA, Michel TOLLENAERE, Aref MAALEJ.

Etude d'un mécanisme de versionnement de connaissances dans une démarche de conception collaborative : Congrès International CMSM'2007 ; 19-21 mars 2007 Monastir-Tunisie : Alaeddine Zouari, Michel Tollenaere, Habib Ben Bacha, Aref Maalej.

Ecole doctorale Organisation Industrielle et Systèmes de Production

L'Ecole Doctorale Organisation Industrielle et Systèmes de Production est une école thématique du site grenoblois. Elle a vocation à coopérer avec toutes les composantes universitaires traitant de l'organisation industrielle, pôle d'excellence du site grenoblois. Elle entend donc encourager les échanges interdisciplinaires, notamment par les co-directions de mémoire ou de thèse. Elle fédère trois filières : Génie industriel (filiale interdisciplinaire pour ingénieurs visant la maîtrise globale des systèmes de production et de la performance industrielle), Sociologie industrielle (analysant les changements en cours dans les organisations de travail), et Economie industrielle (formant aux nouvelles théories de l'économie industrielle et à l'économie du management). Ces filières s'appuient sur un potentiel de huit laboratoires dont sept Unités Mixtes de Recherche rattachées au CNRS et à l'INRA : GILCO, CRISTO, GAEL, 3S, LAG, LEPII, LSR et CERAG. L'ED OISP organise une journée annuelle de présentation des thèses en cours avec la participation active des directeurs de thèse et de laboratoire. Elle s'efforce de contribuer à la diffusion des travaux des doctorants à l'extérieur tant auprès des communautés scientifiques que des milieux professionnels. Elle œuvre à l'insertion professionnelle des docteurs dans les carrières publiques de l'enseignement supérieur et de la recherche et celles du monde économique. Elle propose chaque année aux doctorants et aux enseignants et chercheurs des séminaires pluri-disciplinaires. Elle facilite l'internationalisation de la formation des doctorants.

La formation doctorale en Génie Industriel de Grenoble

La formation doctorale en Génie Industriel de Grenoble s'articule autour de trois thèmes : la conception de produits, la conception et la gestion des systèmes de production, l'innovation industrielle, et développe une approche interdisciplinaire des questions techniques, économiques et humaines posées par les entreprises dans la maîtrise de leur production. La formation s'appuie sur le Master Recherche en Génie Industriel et conduit ainsi à une meilleure prise en compte des problèmes de l'industrie, notamment manufacturière, sur la base de la formation d'origine des étudiants issus des filières EEA, Mécanique, Recherche Opérationnelle et, sous certaines conditions, des Sciences Humaines et Sociales.

Les Laboratoires GILCO et GSCOP

Cette thèse s'est déroulée dans le laboratoire GILCO (Gestion Industrielle, Logistique et COncption), regroupant tous les enseignants-chercheurs de l'ENSGI des domaines SPI et STIC. GILCO a rejoint le laboratoire G-SCOP (Sciences de la Conception, de l'Optimisation et de la Production de Grenoble) créé au 1er Janvier 2007. Ce laboratoire G-SCOP est une FRE CNRS, unité mixte CNRS, INPG et UJF. Il regroupe les forces du site Grenoblois dans les domaines de la conception intégrée de produits, l'optimisation et la conception et la gestion des systèmes de production.

L'Institut de la Production et des Organisations Industrielles

L'Institut de la Production et des Organisations Industrielles (IPI) est une structure inter-établissement INPG et Université Pierre Mendès France, pluriformations du Ministère de l'Education Nationale, de la Recherche et de la Technologie. Créé en 1986, l'IPI se donne pour objectif de promouvoir, développer et soutenir des recherches interdisciplinaires dans le domaine de la Productique et du Génie Industriel à Grenoble. Ses domaines de recherche s'organisent autour de trois thèmes majeurs pour la performance industrielle : la conception intégrée de produits, la conception et la gestion des systèmes de production, l'innovation. Chaque thème est organisé en projets associant deux ou trois disciplines différentes.

Proposition de mécanismes de versionnement et d'agrégation des connaissances de domaine en conception de produits industriels.

Résumé :

L'étude de thèse s'inscrit dans une logique d'ingénierie concurrente et de conception routinière. Elle s'intéresse au versionnement des connaissances en conception en se basant sur les modèles de MULTI (projet MULTI). Dans une première étape, l'étude se focalise dans la structuration et l'affinage des modèles de MULTI, pour valider ces concepts ils sont appliqués sur les systèmes de climatisation automobile. Le deuxième volet consiste à l'élaboration d'une méthode qui met en émergence la collaboration des acteurs impliqués dans un processus de conception en vue de construire dynamiquement une bibliothèque de projet, qui trace l'historique du processus d'ingénierie et capitalise les connaissances liées au produit dans le but de réutiliser ce patrimoine informationnel. La partie cruciale de cette étude consiste à étudier de façon formelle les mécanismes de versionnement et d'agrégation des connaissances de domaine qui sont continuellement modifiés par les acteurs (actualisation, expansion, restriction, ...) au cours du processus de conception, ce qui engendre une grande diversité de connaissances difficile à gérer. Pour pallier ces problématiques, nous proposons un mécanisme permettant d'exprimer, et de gérer les stratégies d'évolution des versions de connaissances en se basant sur la nouvelle théorie de la conception dite théorie C-K. L'étude a conduit au développement d'une maquette informatique appelé «DISCOVER» pour valider l'approche proposée.

Mots-clés : conception routinière, conception collaborative, gestion des connaissances, capitalisation des connaissances, versionnement et agrégation des connaissances, règles d'évolution des versions.

Proposal of domain knowledge versioning and aggregation mechanisms in industrial products design process.

Abstract:

The study of thesis falls under a logic of concurrent engineering and routine design. It is interested to knowledge versioning in design while being based on MULTI models (project MULTI). In a first stage, the study is focused in structuring and refining MULTI models, to validate these concepts they are applied to car air-conditioning systems. The second shutter consists with the development of a method which emergences the collaboration of the actors implied in a design process in order to dynamically build a library of project, which traces the history of the engineering process and capitalizes knowledge related to the product with an aim of re-using this informational inheritance. The crucial part of this study consists in studying in a formal way the domain knowledge versioning and aggregation mechanisms. In design process, knowledge is continuously modified by actors (actualization, expansion, restriction...); this generates great knowledge diversity so difficult to manage. To mitigate these problems, we propose a mechanism allowing to express, and to manage the strategies of evolution of knowledge versions in this basing on a new theory of design known as C-K theory. The study led to the development of a data-processing model called "DISCOVER" to validate the approach suggested.

Key words: routine design, collaborative design, knowledge management, capitalization of knowledge, knowledge versioning and aggregation, rules of versions evolution.