



SI-COTS : Aide à l'intégration de COTS Products

Philippe Roose

► **To cite this version:**

Philippe Roose. SI-COTS : Aide à l'intégration de COTS Products. Workshop MADSI - Inforsid 2008, May 2009, Fontainebleau, France. 2007. <hal-00346867>

HAL Id: hal-00346867

<https://hal.archives-ouvertes.fr/hal-00346867>

Submitted on 20 Apr 2009

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.

SI-COTS :

Aide à l'intégration de COTS Products

Philippe ROOSE

LIUPPA/IUT de Bayonne/UPPA

Château Neuf, Place Paul Bert

64100 Bayonne

Philippe.Roose@iutbayonne.univ-pau

RESUMÉ. Ce papier présente un système d'information permettant d'aider la sélection de COTS Product. Nous présentons une définition de ce qu'est un COTS Product, le système d'information avec les exigences requises pour une intégration efficace et enfin un outil permettant aux développeurs de vérifier la sélection et de réaliser les prototypes nécessaires.

ABSTRACT. This paper presents an Information System based solution in order to help the selection of COTS Products. We present a definition of what is a COTS Product, the IS itself, the requirements in order to achieve an efficient integration and a tool implementing the information system helping developers to check the selection in order to develop prototypes.

MOTS-CLÉS. : Système d'Information, COTS, COTS Product, Prototype

KEYWORDS. : Information System, COTS, COTS Product, Prototype

1 Introduction

La problématique à laquelle nous nous intéressons est celle des applications logicielles réutilisant des produits logiciels préfabriqués : *Commercial Off-The-Shelf Products* ou produits logiciels sur étagère en français. L'idée de réutiliser des produits préexistants existe depuis bien longtemps mais achoppe sur le manque de techniques et d'outils traitant spécifiquement de tels actifs logiciels : comment les sélectionner/évaluer ? Comment les intégrer ? Comment les remplacer/faire évoluer ? Etc. Afin de minimiser ces contraintes, nous proposerons l'utilisation d'un environnement permettant de faciliter la mise en œuvre de prototypes d'expérimentation des COTS Products.

Ce papier présente ici une solution à base d'un système d'information dont l'objectif est de faciliter le recueil d'informations sur chaque COTS préalablement sélectionné ainsi que sur les contraintes d'intégration de manière à évaluer les difficultés, la faisabilité et de mettre en avant les incompatibilités. Dans une première partie nous définirons ce que nous entendons par le terme COTS Product. Dans une seconde partie, nous présenterons l'approche que nous avons utilisée sur la

base d'un SI et d'un prototype permettant l'évaluation. Enfin, nous montrerons ensuite le SI et le prototype basé sur ce SI qui permet l'aide à la sélection.

2 Les COTS

La définition admise par tous est qu'un COTS (*Commercial Off The Shelf*) est un composant issu du marché ou plus communément appelé un composant sur étagère. Derrière cette définition très large se rattache un certain nombre d'attentes.

Ainsi, un COTS est un produit logiciel existant en de multiples copies, dont le code source est disponible ou non, vendu, loué ou fourni gratuitement par un vendeur et ayant des mises à jour périodiques s'accompagnant d'un accroissement des fonctionnalités fournies alors que certaines autres deviennent obsolètes

Cette définition est suffisamment ouverte pour que beaucoup de produits logiciels puissent être considérés comme des COTS. Aussi, nous proposons une définition plus restrictive mais plus cohérente avec les attentes que l'on a d'un COTS Product : **un COTS Product est un produit ayant la capacité de proposer l'un de ses services (ou plus) comme partie indispensable au fonctionnement d'une autre application.**

Le terme générique et tronqué de COTS regroupe en réalité cinq problématiques qui sont autant de domaines de recherche distincts : les *COTS System*, *COTS Based Product*, *COTS Software*, ***COTS Product***, (ce qui nous concerne), *COTS Component*.

La terminologie des *COTS System* [Carney97, Morisio00] considère les COTS comme un ensemble logiciel utilisant des COTS. Dans ce cas, ce n'est pas le produit de manière unitaire que l'on étudie mais le système dans son intégralité.

Le *COTS Software* permet d'identifier l'ensemble des produits. Bien des articles traitent, sous cette dénomination, de la sélection et de la comparaison des produits. Le *COTS Software* est de notre point de vue, une vision plus restrictive que le *COTS System*. Si ce dernier correspond à une vision d'ensemble comprenant l'application à base de COTS, le *COTS Software* ne s'intéresse qu'aux différents COTS indépendamment de l'application à développer.

Une autre restriction dans la vision des COTS correspond à la troisième dénomination *COTS Product* qui s'attarde sur la définition d'un produit et les problématiques relatives à son intégration dans l'application. Dans ce cadre là, la relation, la comparaison et contraintes entre produits ne sont pas abordées comme précédemment. Seules les fonctionnalités ou les questions techniques pour l'intégration sont abordées. Pour finir, les *COTS Components* rassemblent une catégorie de COTS autour des techniques de composants logiciels. C'est une vue particulière des *COTS Product* autour de composants prévus pour être intégrés dès leur conception.

3 Utilisation d'un prototype

L'utilisation de COTS (au sens *COTS Product* dorénavant) dans un processus de développement implique de mettre en œuvre une relation client/fournisseur forte de manière à favoriser la publication des caractéristiques de chaque COTS qui seront utiles et exploitables par les clients.

Selon la norme ISO/IEC 9127 (ISO, 2004), il existe près de 600 conditions de risques ou combinaisons de paramètres d'entrée nécessaires. Ce volume important explique la difficulté du recueil d'informations. De plus, cette quantité importante et la faible structuration de ces dernières provoquent un éparpillement et donc un manque d'accessibilité.

Aussi, les développeurs à base de COTS se tournent fréquemment vers la réalisation de prototypes intégrant un ou plusieurs COTS permettant de :

- ✚ Faire des expérimentations en testant la véracité des explications concernant le COTS au regard de ce qu'il produit/réalise,
- ✚ Recueillir des informations sur ce dernier en termes qualitatifs et quantitatifs, mais également en termes de compatibilité, d'environnement d'exécution, etc.
- ✚ Valider ces informations, en s'assurant par des jeux d'essai de la validité des traitements réalisés.

Toutes ces informations obtenues permettront l'évaluation du COTS seront exploitables afin de faciliter la phase de **sélection**. Néanmoins, la réalisation de prototype se heurte à un certain nombre de difficultés. La première de ces difficultés, et pas la moindre, est d'obtenir les informations nécessaires à l'intégration des COTS. Ces informations ne sont en effet pas forcément disponibles.

La réalisation d'un prototype va permettre de connaître les moyens nécessaires pour intégrer un COTS ainsi que la mise en œuvre de la méthode d'intégration spécifique à chaque COTS.

Si nous souhaitons faciliter le recueil des informations précédemment citées, il est nécessaire de mettre en œuvre une relation « Client / Fournisseur » qui permettra au développeur d'accéder aux données nécessaires à la réalisation de son prototype.

Ainsi, l'obtention de ces informations permettra d'utiliser le COTS indépendamment de son intégration, de pouvoir écrire le code nécessaire à son intégration, de pouvoir faire fonctionner l'application avec l'ensemble des COTS. Ce recueil d'informations se réalise en trois phases distinctes :

Pour l'utilisation des COTS

Informations permettant d'identifier l'ensemble des logiciels nécessaires au fonctionnement du COTS et du service nécessaire dans ce dernier. Par extension à l'ensemble des COTS, nous obtenons un graphe de dépendances.

De la même manière, cette étape permettra l'identification de l'ensemble des logiciels ne permettant pas un fonctionnement optimal identifiant ainsi les incompatibilités plus ou moins inévitables.

Pour intégrer le COTS

Nous identifions ici l'ensemble des logiciels nécessaires au développement de l'intégration, c'est-à-dire, qui permettront l'intégration des COTS dans l'application. Nous retrouvons ici les outils de développement, des logiciels annexes ou complémentaires ainsi que les bibliothèques. C'est ici également que sont identifiés les langages de programmation qu'il faudra manipuler et que les exemples de codes permettant de faciliter l'intégration seront référencés. Nous y trouverons également ce que nous souhaitons utiliser dans l'application.

A un second niveau, c'est à cette étape que sont mises en évidence les configurations ne permettant pas l'intégration de tel ou tel COTS..

Pour certifier les informations

Cette dernière étape permet d'identifier l'auteur des informations. En effet, il est cohérent de penser que le concepteur du COTS lui-même fournit les informations les plus fiables.

Ainsi, le Système d'Information que nous allons proposer permettra de faciliter l'accès aux informations, la mise en relation des clients et des fournisseurs, des traitements automatisés et de vérifier la validité des données. Afin d'éviter toute ambiguïté, il est nécessaire de préciser nous ne proposons pas des processus sur l'évaluation et l'estimation des risques et des coûts. Nous ne proposons pas non plus un système de pilotage pour le traitement des choix de COTS ainsi que pour le traitement des informations recueillies pendant les phases du processus de développement. Nous partons de l'hypothèse que le choix des COTS a été réalisé.

4 Système d'Information

Afin d'établir le SI, nous avons besoin d'un certain nombre d'informations :

- ✚ Un identifiant unique pour les COTS (au même titre que les références uniques des cartes réseaux). Cet identifiant pourrait être par exemple l'identifiant international de l'entreprise fournissant le COTS.

- ✚ Eventuellement une référence à d'autres COTS ou applications (exemple d'un pré-requis ou d'une incompatibilité avec un autre produit).
- ✚ Des informations sur l'utilisation et l'intégration du COTS.

Ces informations sont regroupées pour définir de manière unique le COTS sous la forme d'une carte d'identité ICOTS (*Identity Card for COTS*) (Michel, 05) concernant un ensemble d'informations permettant une caractérisation unique de ce dernier.

4.1 Environnements et scénarios

Nous proposons ainsi la notion d'environnement et de scénario. Pour un environnement d'intégration donné, il existe un ensemble de scénarios valides ainsi qu'éventuellement des scénarios invalides. Chaque scénario est un cas particulier à évaluer. Pour décrire un prototype ou un COTS, nous identifions trois environnements : l'environnement d'exécution, l'environnement d'intégration, l'environnement de développement

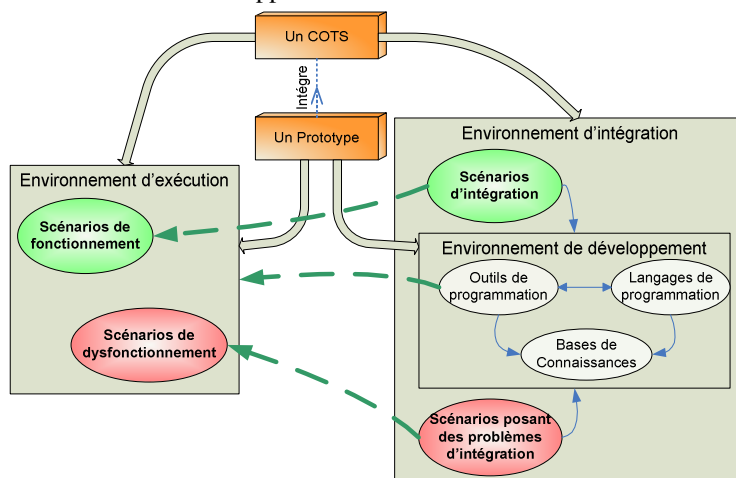


Figure 1 : Trois environnements

Ces informations sont fournies par les concepteurs et fournisseurs de COTS complétées par les acteurs appelés les clients durant le processus de développement d'une application à base de COTS. En fait, elles doivent toutes être rattachées aux produits et donc à la carte d'identité ICOTS. C'est à partir de cette carte d'identité que nous affectons les informations concernant les environnements d'intégration et d'exécution. De ce fait, lorsque les fournisseurs de COTS ont besoin de décrire un

environnement pour leur produit ils peuvent faire référence à une autre carte d'identité qui elle-même décrira son propre environnement. Ainsi, par références successives, il est possible d'obtenir la liste de tous les éléments logiciels nécessaires à un environnement donné et donc de détecter des incompatibilités. Par exemple si dans un prototype nous assemblons les COTS A et B, le premier a besoin pour son exécution d'un produit « Y » qui utilise un produit « Z » mais incompatible avec l'utilisation du produit « X ». Le second a besoin d'un produit « V » utilisant le produit « X ». Dans cet exemple nous pourrions conclure que l'utilisation simultanée du COTS A et B n'est pas possible : c'est donc une incompatibilité que le développeur devra résoudre. Pour cela, il peut choisir d'utiliser l'un ou l'autre des COTS et de développer les services non remplis par le COTS manquant, ou alors de remplacer l'un d'entre eux.

Un ensemble de rapports va aider le développeur à maîtriser les multiples informations issues des différents produits qui vont composer son application. Certaines l'aideront dans la réalisation de l'intégration (exemple du scénario d'intégration) alors que d'autres l'avertiront sur des configurations d'exécution ou de développement ayant déjà causées des problèmes à d'autres utilisateurs de COTS. Il est évident que cela ne peut que favoriser l'utilisation des prototypes pour évaluer, qualifier et estimer les coûts et les risques.

Rapport sur l'environnement d'exécution

- ✚ Ensemble des logiciels nécessaires au fonctionnement de chaque COTS
- ✚ Liste des logiciels incompatibles
- ✚ Les OS compatibles et communs à tous les COTS.

Rapport sur l'environnement d'intégration

- ✚ Liste des logiciels nécessaires pour l'intégration
- ✚ Les services utilisés des COTS et ceux restant à développer
- ✚ Les outils de développement pour l'intégration
- ✚ Les documentations associées à chaque service de COTS

Rapport sur les dysfonctionnements

- ✚ Liste des incompatibilités pour chaque COTS
- ✚ Identification du COTS qui est la cause d'une incompatibilité dans l'assemblage.

4.2 Les acteurs du SI

Le fournisseur et le concepteur de COTS (FourCo)

Dans un circuit de distribution classique on doit dissocier le fabricant d'un produit et le circuit de distribution ou vendeur de ce produit. C'est effectivement, dans bien des cas, deux acteurs différents. Pourtant, dans la relation que l'on essaye de montrer, cette différence n'apporte rien sur le fait que ce soit l'un ou l'autre qui fournira le produit et les informations dont ont besoin les clients. Seule la différence de services proposés entre divers circuits de vente pourrait nécessiter que l'on définisse deux acteurs séparés. Or ce critère là, s'il est important lors de la sélection du vendeur de COTS, n'a pas d'influence sur la qualité et le choix du produit. Ce dernier est identique quel que soit le vendeur. Notre préoccupation est focalisée sur les informations dont ont besoin les clients pour mener à bien leurs missions de sélection et d'intégration des COTS. En particulier nous voulons surtout les aider à réaliser un prototype, et pas spécialement dans la sélection d'un vendeur.

Pour cela, nous ne considérerons ici qu'un seul acteur (appelé *FourCo*) qui sera le responsable des informations que le client pourrait lui demander au titre de concepteur et diffuseur de COTS. Mais dans un premier temps, pour pouvoir initier une relation client/fournisseur, il est indispensable que les FourCo diffusent de l'information sur le produit qu'ils mettent à disposition des futurs clients.

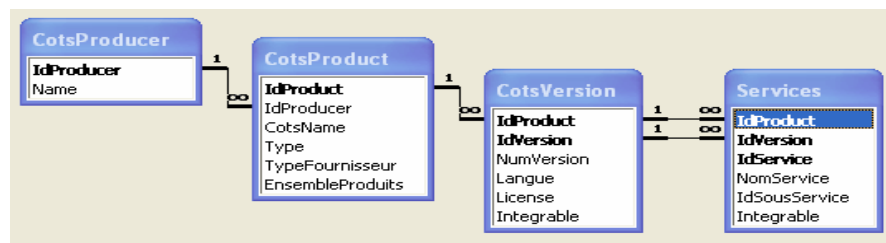


Figure 2 : Schéma E/A de la carte d'identité ICOTS

Cette figure correspond à la base de données du SI qui permet de saisir le minimum d'informations que l'on trouve sur les sites Web des distributeurs de COTS, à savoir les services proposés par chaque produit.

L'utilisateur de COTS

Il doit récupérer de nombreuses informations en partie disponibles chez les fournisseurs de COTS. Pour la partie restante, ils vont devoir les rechercher à travers des évaluations et des tests réalisés à l'aide de prototypes. C'est donc « l'utilisateur de COTS » qui, en lançant la première activité du processus de développement (la

veille technologique), va initialiser un processus de communication avec FourCo qu'il pourra continuer à utiliser tout au long du processus de développement de son application. Il pourra également se faire aider par un nouvel acteur appelé la communauté de COTS.

La communauté de COTS

Elle représente l'ensemble des développeurs avec au moins une expérience de développement d'application à base de COTS. Ils ont acquis de l'information pendant leur développement et notamment durant les phases de qualification, d'adaptation et de réalisation de prototypes, mais aussi pendant les évaluations des risques et des coûts. Plus encore, l'historique de leurs développements permettra de comparer les estimations avec la réalité du coût de l'intégration du COTS. Cela devrait affiner la propre estimation des coûts et risques du projet menée par l'utilisateur de COTS.

La communauté de COTS devient au même titre que les concepteurs des fournisseurs d'informations. Ces acteurs ont un rôle important par le fait qu'ils vont enrichir les fiches descriptives des COTS. Les informations ainsi ajoutées pourront servir à d'autres utilisateurs qui posséderont une information plus complète sur l'intégration des produits.

Afin d'alimenter le SI, nous distinguons différents types d'acteurs. Bien évidemment, nous retrouvons la séparation producteur/consommateur. Néanmoins, les producteurs se scindent en deux parties, selon que ce sont des fournisseurs de COTS (et donc a priori avec un niveau de confiance élevé) ou la communauté des utilisateurs de COTS.

Chaque acteur possède ainsi un rôle de producteur ou de consommateur, sachant qu'à un instant donné un utilisateur peut se retrouver dans le rôle de communauté s'il participe à la production d'informations relatives à l'intégration et donc à l'enrichissement des informations du SI permettant ainsi une capitalisation des informations que va permettre l'application que nous présentons dans le paragraphe suivant.

5 Environnement de saisie

L'obtention des informations est primordiale pour le bon déroulement du processus de développement de l'application basée COTS. Pour réduire les difficultés sur le sujet (Leung, 02) propose de réaliser des prototypes qu'il juge indispensables à plusieurs phases du processus de développement pour le recueil et la validation des informations. Le prototype doit permettre d'étudier l'intégration d'un ou de plusieurs COTS cohabitant ensemble. De cette étude les développeurs vont obtenir et valider des informations indispensables pour chaque phase du

processus de développement de l'application à base de COTS. L'utilisation de ces prototypes dans des expérimentations est une démarche qui garantit l'accès aux informations manquantes et difficilement accessibles. Indépendamment de savoir si cela représente le seul moyen d'obtenir ces informations, le fait est que l'expérimentation est souvent le meilleur moyen pour s'assurer de la validité des informations recueillies. L'application présentée ci-après va mettre en pratique les apports proposés ci-dessus.

5.1 Présentation

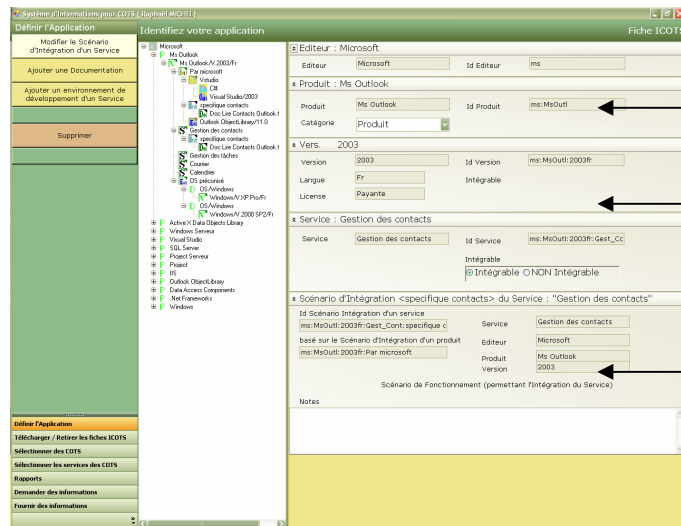
L'application SI-COTS va permettre l'enregistrement de la carte d'identité avec les informations associées à la fois pour le COTS Product mais également pour l'application à base de COTS. Cet outil va également aider à la relation client/fournisseur en mettant l'ICOTS à disposition et en permettant le téléchargement d'ICOTS supplémentaires.

SI-COTS va également permettre l'identification des services des COTS Products utilisés dans l'application à base de COTS, et notamment les services intégrables (ceux qui ne le sont pas ne sont d'aucune utilité). De plus, elle propose pour chaque COTS et service, l'identification des environnements d'exécution et d'intégration, dans le but d'aider le développeur de prototypes, et notamment la saisie des pré-requis, des outils de développement, des exemples de codes et les dysfonctionnements identifiés.

Enfin, dans le domaine du traitement et de la production d'informations, SI-COTS va permettre d'identifier l'ensemble des logiciels nécessaires à l'exécution du prototype, l'ensemble des logiciels nécessaires au développement du prototype et l'intégration des COTS et enfin, les dysfonctionnements éventuels dans le cadre de l'utilisation des COTS dans le prototype. Le modèle de données est disponible dans (Michel, 2006).

A titre d'illustration, nous allons utiliser deux COTS Products bien connus, MS Outlook et MS Project afin d'utiliser respectivement les services de gestion des contacts et des tâches.

L'utilisation de SI-COTS se réalise en 4 étapes. La première (cf. Figure 3) est réalisée par le fournisseur de COTS, c'est la création de la fiche ICOTS (ici de Ms Outlook). Les trois étapes suivantes sont réalisées par l'utilisateur et correspondent à la création de la fiche ICOTS dans le prototype et de l'association des services et la production des rapports (cf. figure 4).



La fiche ICOTS
 Environnement
 d'exécution
 Scénario
 d'intégration

Figure 3 : Fiche ICOTS (coté fournisseur)

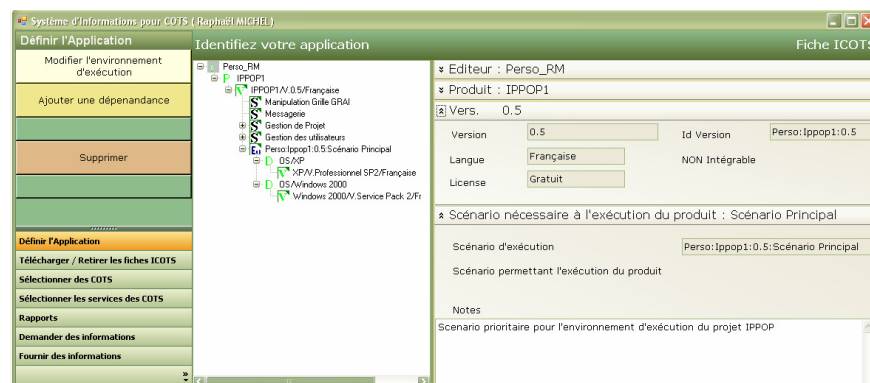


Figure 4 : Fiche ICOTS (coté utilisateur)

6 Conclusion

Le prototype développé en .NET pour plateforme Windows est téléchargeable à l'URL suivante (<http://www.iutbayonne.univ-pau.fr/~roose/pub/recherche/SICOTS>). Il permet l'amélioration du recueil des informations et facilite l'accès aux informations pour l'intégration ainsi qu'un gain de temps important. De plus il favorise les évaluations par la production de rapports et permet une capitalisation des informations via la communauté des utilisateurs de COTS Products. Le Système d'Information développé et implémenté dans l'application SI-COTS propose une aide aux développeurs de prototypes d'applications à base de COTS Products. Cette approche par l'intermédiaire d'un SI a abouti à la définition d'une carte d'identité

(ICOTS) ainsi qu'à la production d'informations d'aide à l'intégration de COTS Products. L'accès aux informations étant rare et précieuse, il est important de capitaliser l'expérience acquise au gré des développements, ce que permet l'application développée. Néanmoins, l'approche possède ses limites en termes d'interopérabilité entre ICOTS qui nécessite une normalisation des échanges et des données. De plus, nous n'avons abordé l'approche des COTS Products que par l'aide à l'intégration, ce qui ne couvre bien évidemment pas le spectre très large des phases amont de sélection, évaluation, qualification et d'estimation des risques et des coûts.

7 Bibliographie

(Carney, 03) David J. Carney, Patricia A. Oberndorf, Patrick R.H. Plac - A Basis for an Assembly Process for COTS-Based Systems (APCS) ; Technical Report ; CMU/SEI-2003-TR-010, ESC-TR-2003-010 ; *May 2003*.

(Leung, 02) Karl R.P.H. Leung, Hareton K.N. Leung - On the efficiency of domain-based COTS product selection method: Information and Software Technology pp 703-715; 44 (2002).

(Michel, 05), Michel Raphaël, Roose Philippe, Barbier Franck - *Information System for Evaluation of COTS* - 3rd International Conference on Software Engineering Research, Management and Applications (SERA2005) - Central Michigan University, Mount. Pleasant, Michigan, USA - IEEE Computer Society 2005, ISBN 0-7695-2297-1, pp.64-69 - August 11 – 13 - 2005

(Michel, 06) Michel Raphael - *Système d'Information pour l'Évaluation des Composants Logiciels sur étagère – (COTS Components)* – Thèse de Doctorat, UPPA, Décembre 2006

(Morisio, 00) Maurizio Morisio, Nancy Sunderhaft - Commercial-Off-The-Shelf (COTS): A Survey A DACS State-of-the-Art Report ; December 2000.

(ISO, 2004) Software engineering -- Product quality -- Part 4: Quality in use metrics - SO/IEC TR 9126-4:2004