

Chapter 5 Common Warehouse Metamodel(CWM)

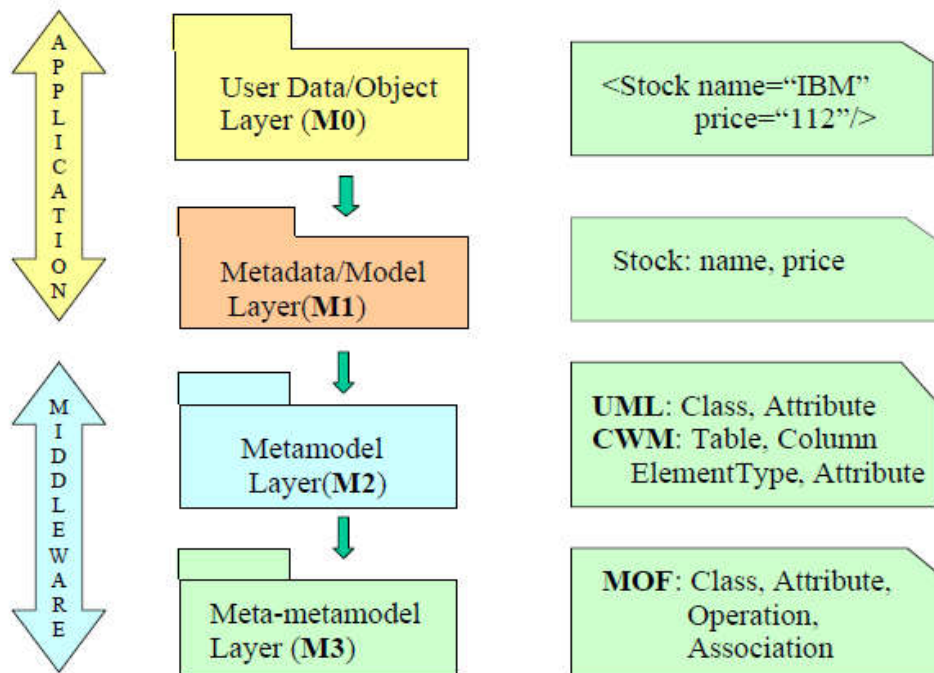
1. Introduction

L'OMG voulait faire de CORBA middleware, mais il est loin d'être le seul à être utilisé. D'autres comme EJB ou .NET sont largement répandus. C'est en partant de cette observation que l'OMG s'est engagé sur la voie du MDA, afin de résoudre les problèmes d'interopérabilité et de portabilité dès le niveau de modélisation. Le MDA se veut donc indépendant de toute plate-forme et de tout système, il permet de concevoir des applications portables au niveau des langages de programmation, des systèmes d'exploitation mais aussi des middlewares (Un middleware est un logiciel qui fournit aux applications des fonctionnalités et des services communs. Il joue le rôle de lien entre les applications, les données et les utilisateurs).

2. Architecture de méta-données d'OMG

Le Model Driven Architecture (MDA) est une démarche de développement proposée par l'OMG. Elle permet de séparer les spécifications fonctionnelles d'un système des spécifications de son implémentation sur une plate-forme donnée. A cette fin, le MDA définit une architecture de spécifications structurée en modèles indépendants des plates-formes (PIM) et en modèles spécifiques (PSM).

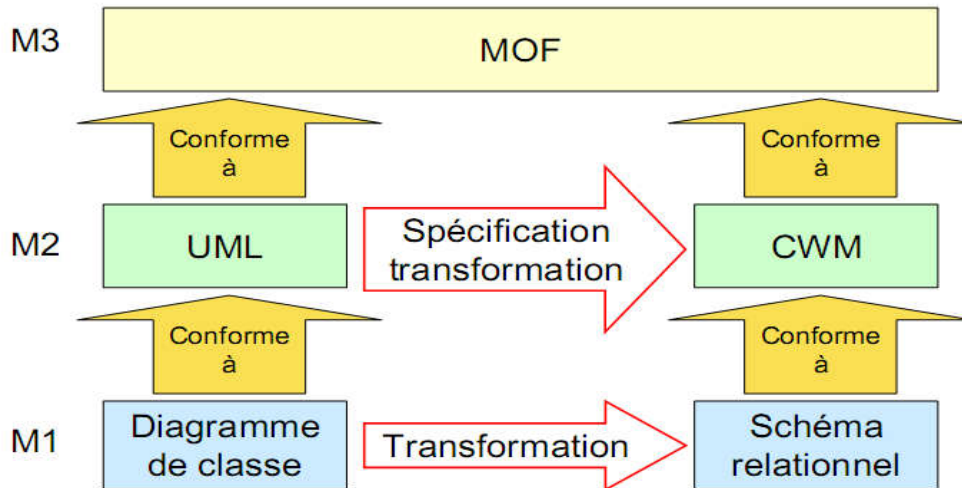
Niveau *	Limites de MOF	Exemples	Utilisation typique
M3	meta-metamodel	The "MOF" Model	Réalisateur de normes
M2	metamodel, meta-metadata	UML metamodel, CWM metamodel	Réalisateur d'outil de logiciel
M1	model, metadata	UML models, CWM metadata	Créateur d'application
M0	object, data	Systèmes modélisés, données d'entrepôt	Utilisateur



Architecture de métamodélisation OMG

3. Transformations de modèles

Les techniques de transformation de modèles sont donc au cœur du MDA. L'OMG doit pouvoir les formaliser et les standardiser pour garantir la compatibilité entre les outils MDA. Afin de pouvoir générer le code fonctionnel, il est également possible d'enrichir les modèles PIM avec un langage d'actions. De plus, les modèles peuvent être accompagnés de contraintes en OCL (Object Constraint Language) afin de vérifier leur validité tout au long du processus. A terme, l'OMG souhaite pouvoir automatiser la plus grande partie de ces transformations (voire figure 4).



Transformation des méta données

4. Pourquoi le standard CWM ?

La quantité de données dans une organisation indiquée double tous les cinq ans. La plupart des organismes souffrent d'une surabondance de données redondantes et contradictoires il est difficile de contrôler pertinemment, accéder, et utiliser que pour la prise de décision.

- Beaucoup de bases de données
- Beaucoup de dépôts
- Beaucoup de schémas décrivant les « mêmes » données
- Les données mobiles exigent la transformation manuelle de schéma

L'entrepôt de données fournit une excellente approche pour des données de transformation dans l'information utile et fiable pour supporter le processus décisionnel d'affaires et pour réaliser l'intelligence d'affaires. Un des aspects les plus importants de l'entrepôt de données est des méta-données.

Puisque chaque outil de gestion des données et d'analyse exige les différentes méta-données et différentes méta-données modèlent (connu comme metamodel) pour résoudre l'entrepôt de données, il n'est simplement pas possible d'avoir un dépôt simple de méta-données qui met en application un méta model simple pour toutes méta-données dans une organisation. Au lieu de cela, ce qui est nécessaire est une norme pour l'échange des méta-données d'entrepôt.

Le CWM est une réponse à ces besoins, il fournit un cadre pour représenter des méta-données au sujet des points d'émission, les cibles de données, les transformations, et l'analyse, et les processus et les exécutions qui créent et contrôlent des données d'entrepôt et fournissent des informations de lignée au sujet de son utilisation.

4.1. Solution CWM

Pour résoudre le problème de gestion et d'intégration de méta-données CWM doit fournir :

- Un langage standard pour définir la structure et la sémantique des méta-données d'une voie formelle
- Un mécanisme standard d'échange pour partager des méta-données définies dans le langage standard
- Un cahier des charges commun qui définit, dans le langage standard, la structure et la sémantique des méta-données partagées dans les données entreposant et l'intelligence d'affaires

Donc le CWM

-Permet l'échange facile de

- entrepôt et
- intelligence d'affaires

- méta-données entre

- outils d'entrepôt
- plateformes d'entrepôt et
- dépôts de méta-données

Dans les environnements hétérogènes distribués.

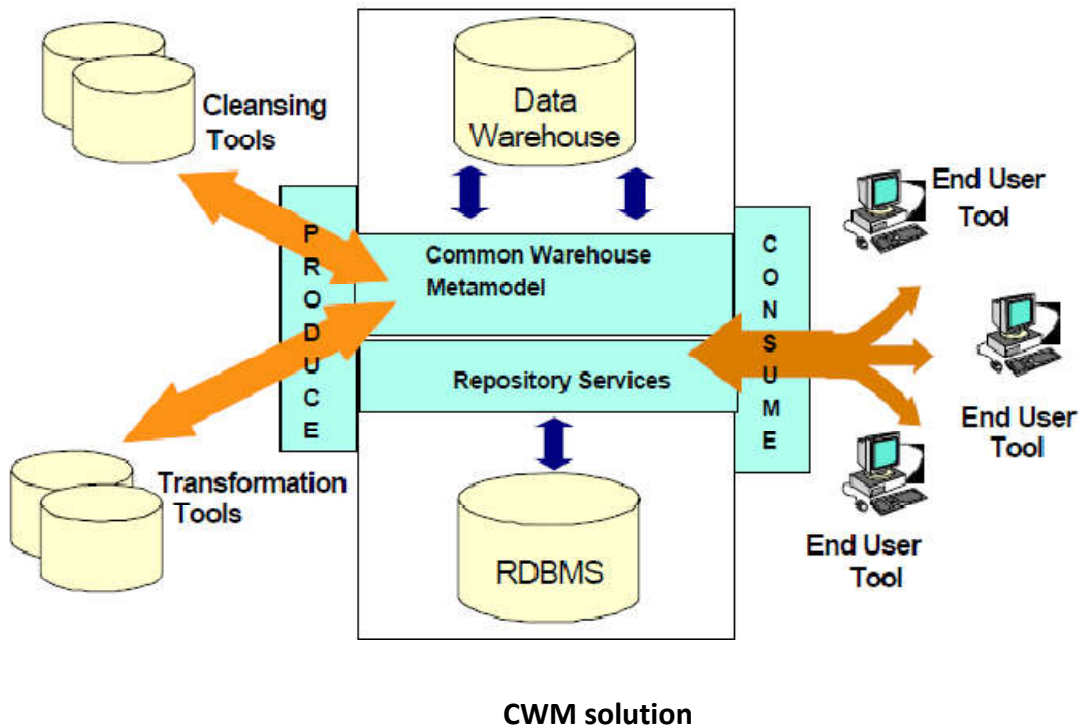
5. Common Warehouse Metamodel(CWM)

5.1 Définition du CWM

CWM est le standard de l'OMG qui traite des entrepôts de données. Il couvre le cycle de vie complet de modélisation, construction et gestion des entrepôts de données.

L'approche préconisée par ce standard pour la migration entre systèmes hétérogènes est une approche MDA (la création de modèles et leurs transformations). CWM définit un méta-

modèle qui représente les méta-données aussi bien métiers que techniques qui sont le plus souvent retrouvées dans les entrepôts de données.



NB : cleaning tools :outil de nettoyage, transformation tools ; outil de transformation,end user tool : outil de l'utilisateur final, repository service : service de référentiel.

CWM définit actuellement les méta-modèles des principaux types d'entrepôts de données (Relationnel, Objet, XML,...) et propose des règles de transformation entre ceux-ci.

5.2 But de CWM

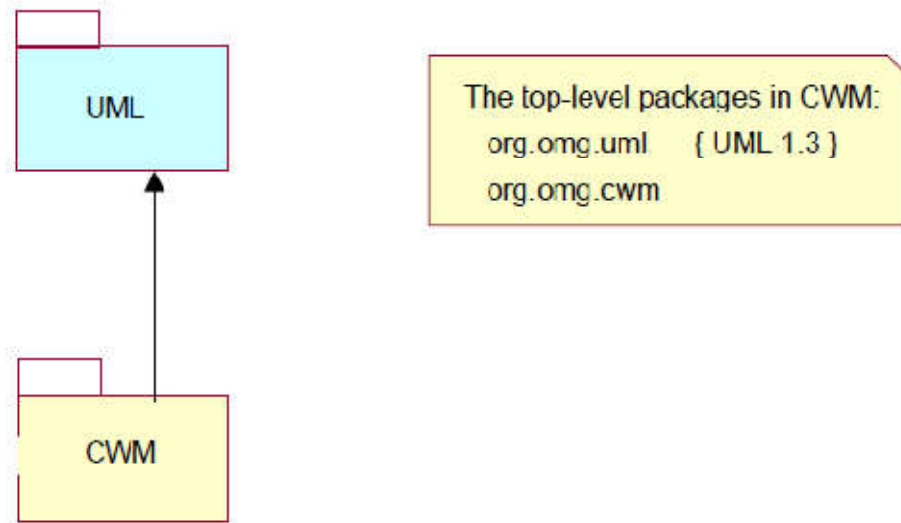
Le but principal de CWM est de permettre l'échange facile de l'entrepôt et des méta-données d'intelligence d'affaires entre les outils d'entrepôt, les plateformes d'entrepôt et les dépôts de méta-données d'entrepôt dans les environnements hétérogènes distribués.

CWM est basé sur trois normes d'industrie principale :

- UML - Unified Modeling Language, un OMG modelant la norme
- MOF - Meta Object Facility, un OMG metamodeling et norme de dépôt de méta-données
- XMI - XML Metadata Interchange, une norme d'échange de méta-données d'OMG

6. Raisonnement de conception

6.1 Réutilisation des concepts d'UML



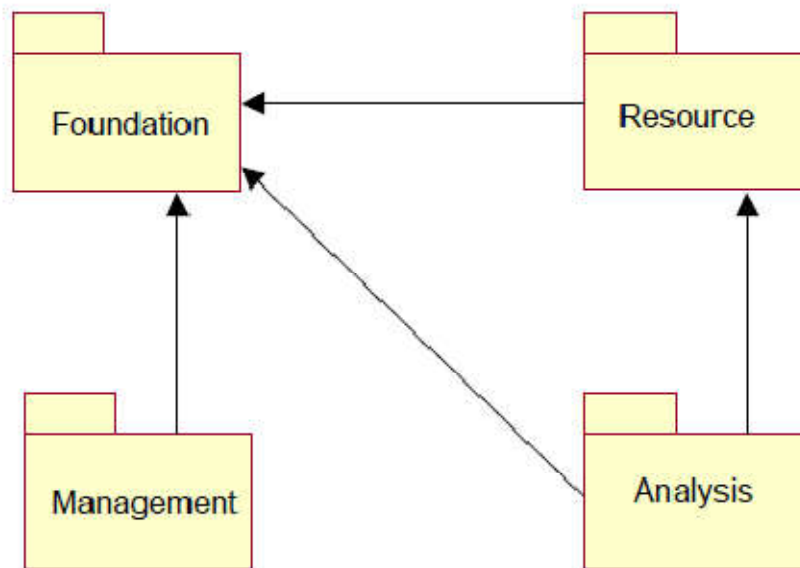
CWM – Niveau supérieur

Le metamodel de CWM a en tant que sa base un modèle d'objet basé sur une version du metamodel d'UML dans lequel ces aspects qui ne sont pas appropriés dans un scénario d'entrepôt de données ont été retirés. Le metamodel de CWM est établi sur et étend ce modèle d'objet.

Plusieurs des types et des associations d'objet du noyau UML sont reflétées par le modèle d'objet de CWM. Partout où approprié, les types modèles d'objet sont subtyped pour fournir un objet plus spécifique saisis le metamodel de CWM, normalement avec des attributs ou des associations supplémentaires. Tous les types d'objet de CWM sont des sous-types directs ou indirects des types appropriés de modèle d'objet, et ainsi héritez de leurs attributs et associations.

Cette approche a beaucoup d'avantages. Elle permet au cahier des charges de CWM de profiter de l'investissement substantiel pour développer et raffiner le metamodel d'UML. La conscience générale des concepts d'UML devrait faciliter l'arrangement du cahier des charges de CWM et de son modèle de base d'objet. Elle permet également l'inclusion facile des modèles d'UML en tant qu'élément des méta-données d'entrepôt de données.

6.2. Modularité



Aperçus CWM

Le méta modèle de CWM est fractionné dans un ensemble de modules. Ceci facilite la compréhension du méta modèle en le dédoublant vers le haut dans de plus petites unités, et permet à des utilisateurs et à des applicateurs d'ignorer les modules qui ne sont pas appropriés à leurs besoins.

Le metamodel de CWM a une structure posée :

- La base comprend le modèle UML-basé d'objet et le CWM Base, qui supporte les concepts et les structures supplémentaires qui sont partagés par d'autres modules. Supplémentaire, le module de déploiement de logiciel supporte l'information de déploiement pour les points d'émission et les cibles dans la prochaine couche.
- Les modules apparentés, record, multidimensionnels, et de XML supportent la définition de divers types des points d'émission et de cibles de données.
- La transformation, les OLAP(traitement analytique en ligne), l'exploitation de données, la visualisation de l'information, et les modules de nomenclature d'affaires définissent les transformations et analytique le traitement de cela a lieu sur ces points d'émission.

- En conclusion, le module de processus d'entrepôt supporte l'information d'établissement du programme, et le module d'exécution d'entrepôt est employé pour enregistrer les détails opérationnels tels que les résultats des passages de transformation.

6.2. Modèle générique

Beaucoup d'attention a été prise pour s'assurer que le méta model de CWM a été rendu aussi générique que possible, et que seulement l'information qui est en commun entre différentes réalisations a été incluse dans le metamodel. Shareability d'information a été contrôlé et raffiné en examinant les besoins de méta-données de plusieurs différents, mais du représentant, réalisations aussi bien qu'un large intervalle des configurations représentatives d'entrepôt.

7. Importance de CWM

- Le consensus industriellement compatible pour des données entrepose l'échange et l'interopérabilité
- Permet l'intégration d'information d'entreprise de sorte que nous puissions améliorer la puissance les données que nous avons
- Supporte déjà la base de données et les systèmes de fichiers les plus employés couramment
- Extensible aux besoins de détail de constructeur/client
- Constructions sur UML, XMI et MOF
- Accélère le marché d'entreposage de données, portique et d'affaires d'intelligence

8. Description du CWM

8.1 L'ensemble de CWM metamodel :

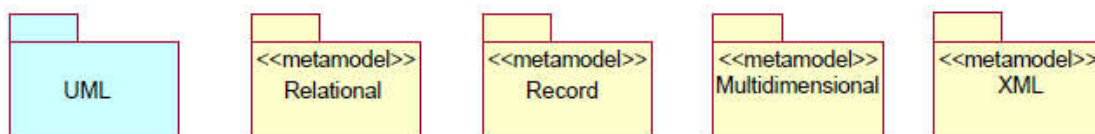
The CWM Metamodel

Management	Warehouse Process			Warehouse Operation		
Analysis	Transformation		OLAP	Data Mining	Information Visualization	Business Nomenclature
Resource	Object Model	Relational	Record	Multidimensional		XML
Foundation	Business Information	Data Types	Expression	Keys and Indexes	Type Mapping	Software Deployment
Object Model						

Common Warehouse Métamodèle

Foundation package

- Business Information package: contient des classes et des associations qui représentent informations commerciales sur les éléments du modèle.
- Data Types package: contient des classes et des associations qui représentent des constructions, que les modélisateurs peuvent utiliser pour créer les types de données spécifiques dont ils ont besoin.
- Expressions package : contient des classes et des associations qui représentent l'expression des arbres.
- Keys and Indexes package : contient des classes et des associations qui représentent des clés et des index.
- Software Deployment package : contient des classes et des associations qui représentent comment le logiciel est déployé dans un entrepôt de données.
- Type Mapping package: contient des classes et des associations qui représentent le mappage des types de données entre différents systèmes.



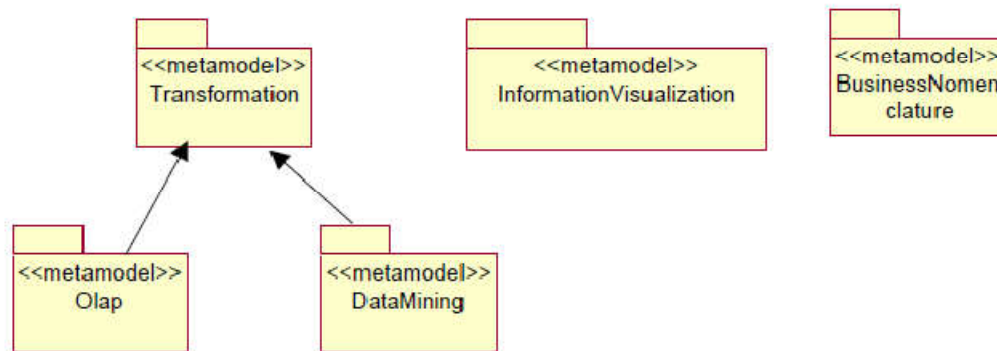
Ressources CWM

Resource package

- Relational package: contient des classes et des associations qui représentent les métadonnées de ressources de données relationnelles.
- Record package : contient des classes et des associations qui représentent les métadonnées de enregistrer les ressources de données.
- Package multidimensionnel : contient des classes et des associations qui représentent

métadonnées de ressources de données multidimensionnelles.

- XML package: contient des classes et des associations qui représentent les métadonnées de Ressources de données XML.



CWM Analyse

Analysis package

- Transformation package: contient des classes et des associations qui représentent métadonnées des outils de transformation de données.
- OLAP package: contient des classes et des associations qui représentent des métadonnées de données en ligne, outils de traitement analytique.
- Data Mining package: contient des classes et des associations qui représentent des métadonnées, d'outils d'exploration de données.
- Information Visualization package : contient des classes et des associations qui représentent les métadonnées des outils de visualisation d'informations.
- Business Nomenclature package - Contient des classes et des associations qui représentent les métadonnées sur la taxonomie des entreprises et glossaire

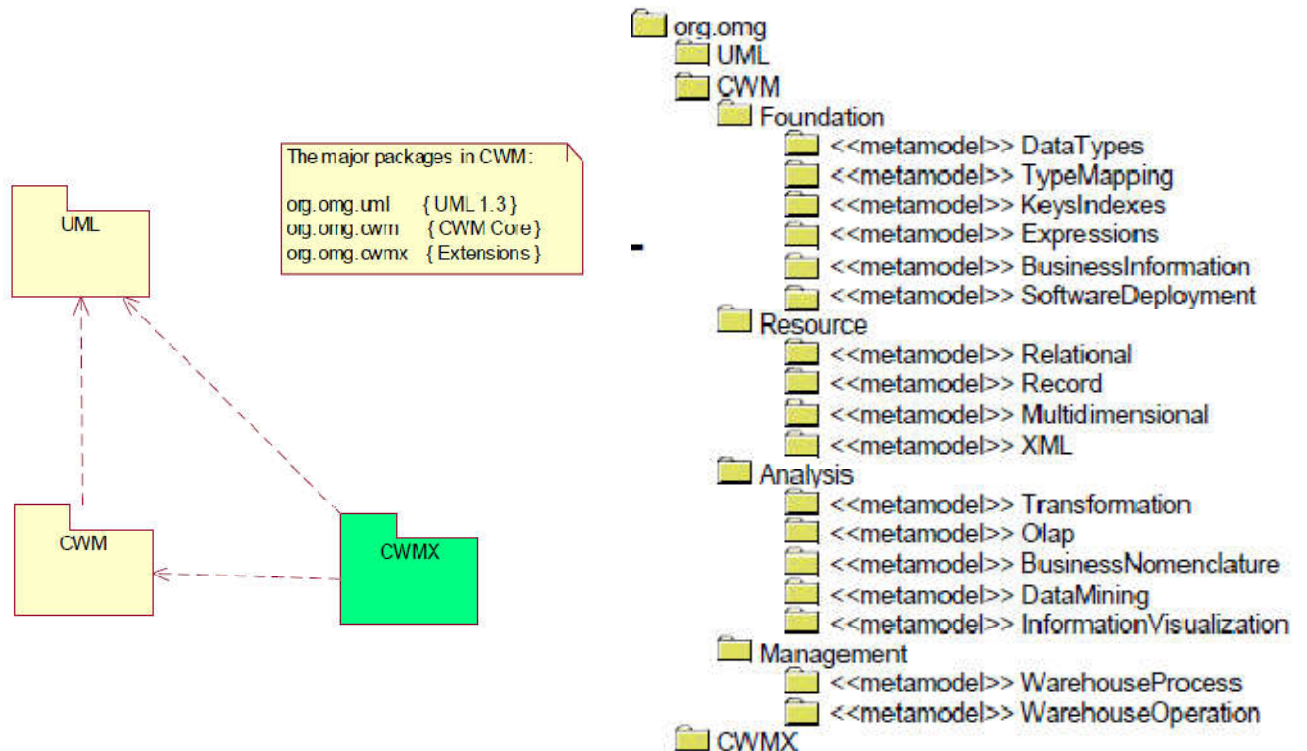
Management package

- Warehouse Process package: contient des classes et des associations qui représentent métadonnées des processus d'entrepôt.
- Warehouse Operation package : contient des classes et des associations qui représentent métadonnées des résultats des opérations d'entrepôt.

UML est la base modelante sur laquelle CWM est construit. Dans la mesure du possible, les Co-présentateurs de CWM ont directement réutilisé les classes existantes et les associations plutôt que de créer d'UML des versions CWM-spécifiques d'elles. Ce choix réduit le nombre de nouvelles classes et d'associations de CWM et accroît les qualifications existantes des modeleurs UML-bien informés.

9. Les package de CWM

Les modules d'utilisations de CWM Metamodel et une structure hiérarchique de module pour contrôler la complexité, pour la favoriser entendre, et pour supporter la réutilisation.



Les couches du CWM décrivent les modules de metamodel qui adressent le cycle de vie complet des applications d'entrepôt de données.

10. CWM et MOF

Le MOF a été adopté comme norme d'OMG pour représenter des metamodels. Le metamodel de CWM a été conçu pour se conformer à cette norme. Ceci permet à CWM d'utiliser d'autres caractéristiques d'OMG qui dépendent du MOF. En particulier, il permet l'utilisation de XMI d'échanger des méta-données d'entrepôt qui est représentée en utilisant le metamodel de CWM, et il permet l'utilisation d'IDL (et d'autres langages de programmation) pour l'accès selon les programmes aux méta-données d'entrepôt basées sur le metamodel de CWM.

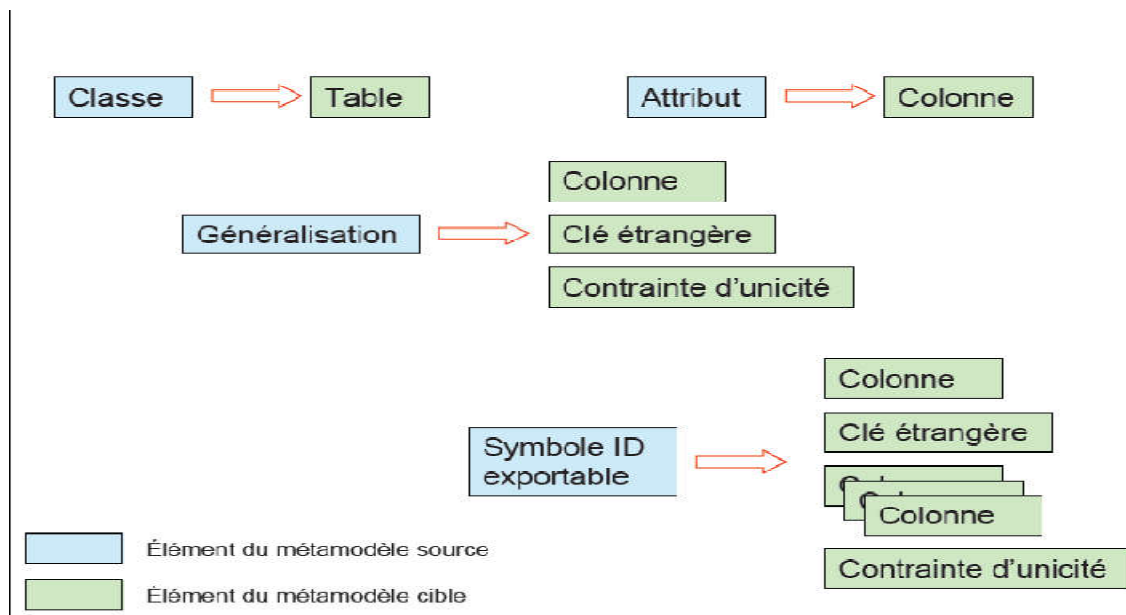
11. CWM et UML

Un premier objectif du CWM est de définir un méta modèle (ou, d'une manière équivalente, un « modèle de méta données ») d'une architecture générique d'entrepôt de données. Ainsi, le CWM méta modèle définit des règles formelles pour modeler des exemples

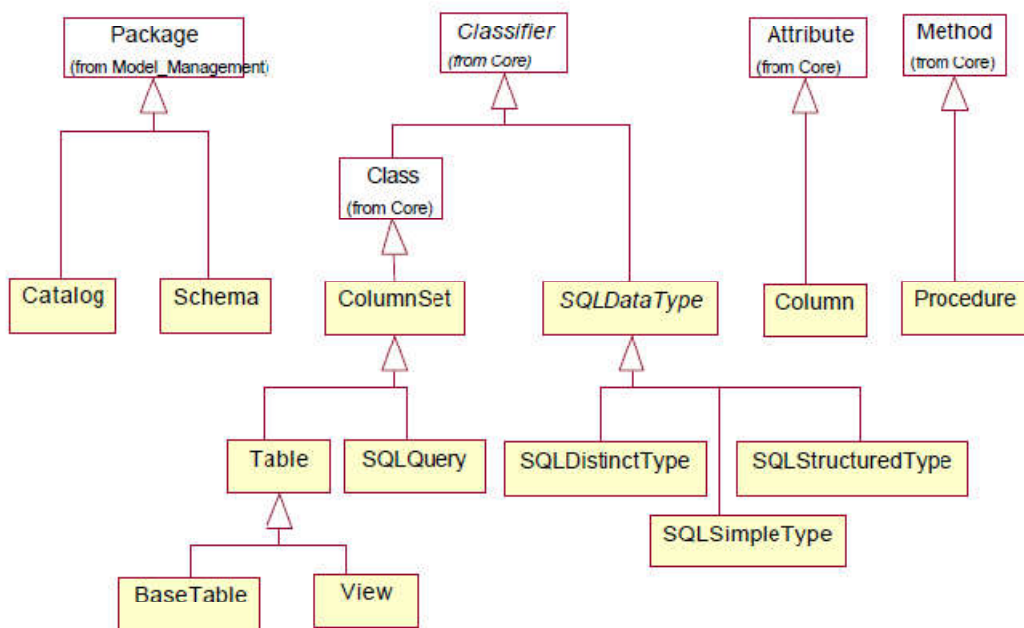
des entrepôts de données. Cependant, il y a également une condition pour le metamodel de CWM d'être exprimée en MOF.

La spécification d'UML est également utilisée dans les voies suivantes :

- La notation d'UML est utilisée dans les représentations schématiques du CWM méta model.
- Des contraintes supplémentaires sur le méta model de CWM sont représentées en langage de contrainte d'objet (OCL), comme défini dans la spécification d'UML.



UML vers CWM



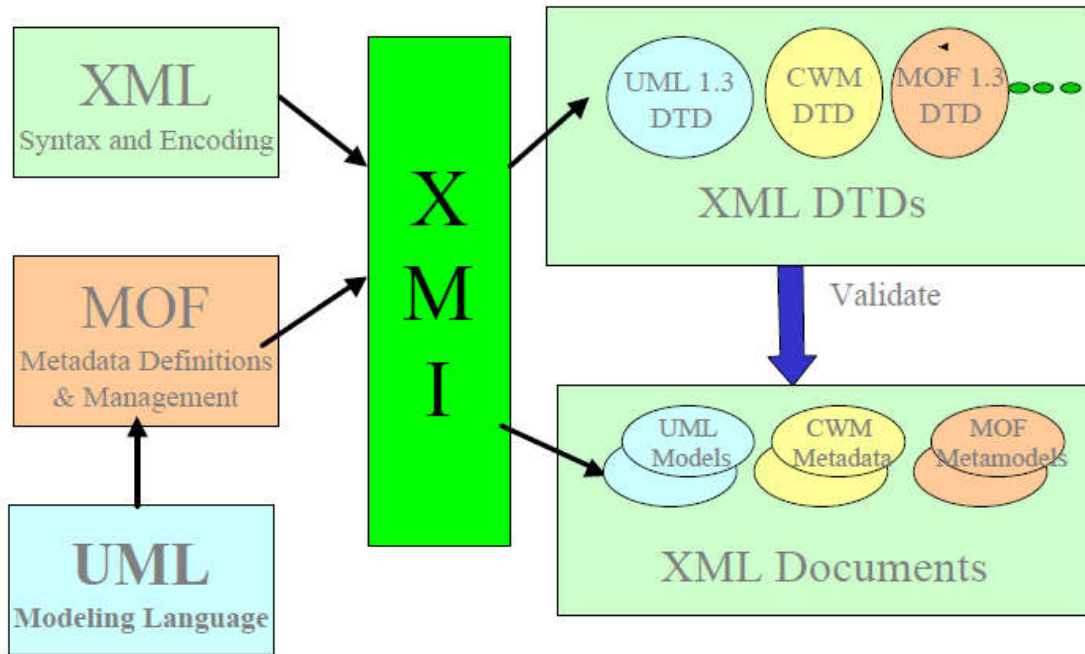
Métamodèle relationnel

12. CWM et XMI

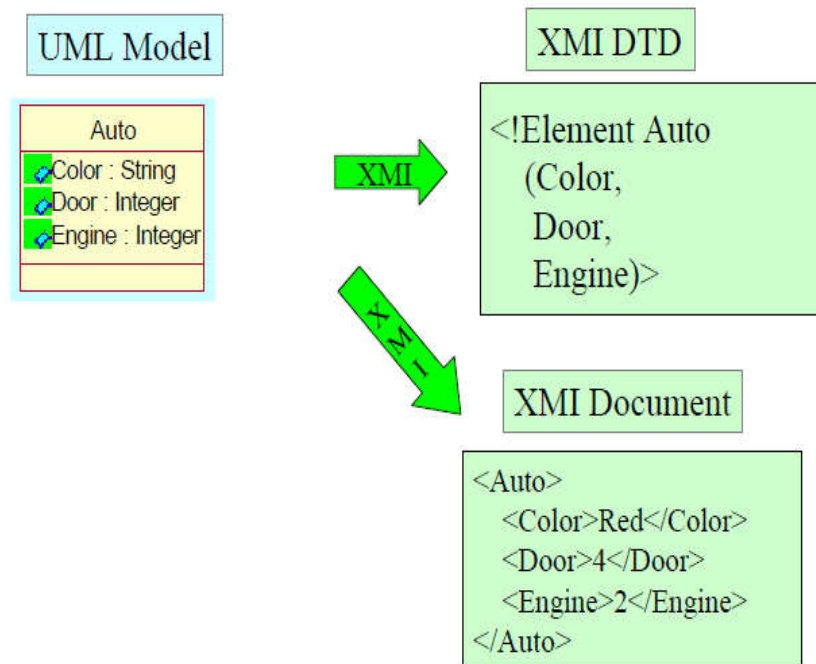
CWM utilise XMI en tant que son mécanisme d'échange. Ceci signifie que toute puissance et la flexibilité de XMI est disponible pour échanger les méta données d'entrepôt et le méta model de CWM elle-même. CWM n'exige aucune extension à XMI.

Un DTD standard pour le méta model de CWM est produit en utilisant les règles de production du DTD de XMI. Des méta données d'entrepôt peuvent alors être encodées comme un document de XML en utilisant les règles de production du document de XMI.

Un document standard de XML pour le méta model de CWM est également produit en utilisant les règles de production du document de XMI, basées sur le DTD de MOF.



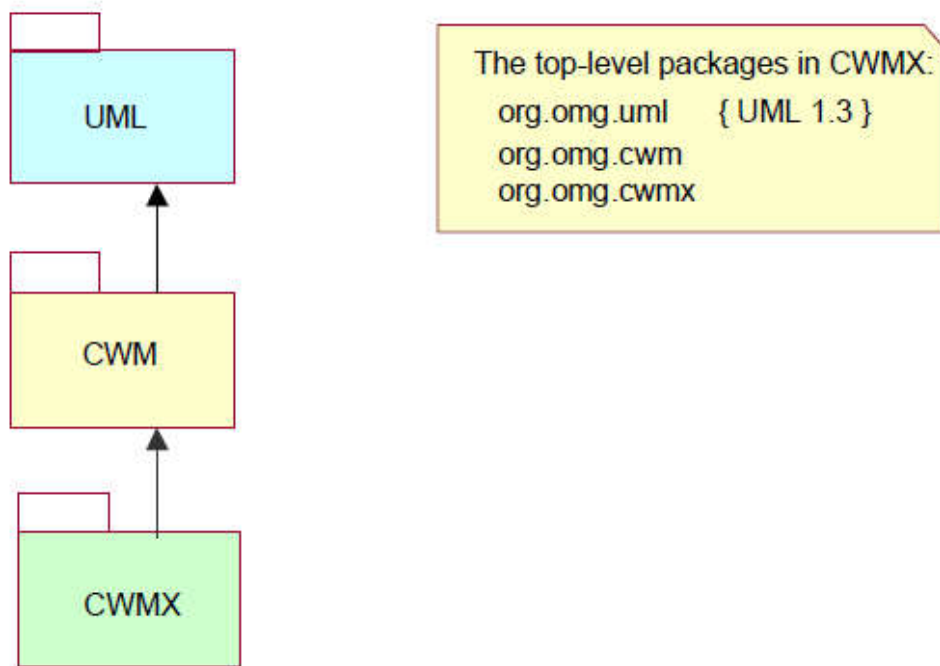
XMI simplifié



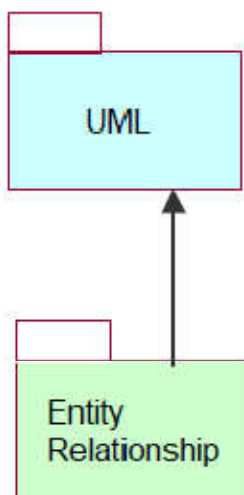
XMI - Automobile Example

13. Extensions de CWM (CWMX)

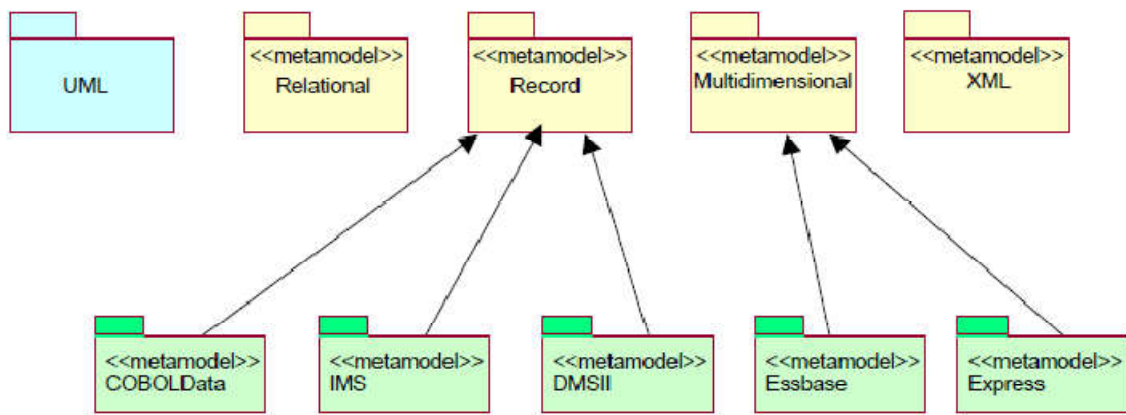
- Metamodel spécifique édité de constructeur afin d'échange de méta-données.
- Ascendance commune dans le metamodel de CWM
- Démontre la validité du metamodel de CWM
- Démontre l'extensibilité du metamodel de CWM



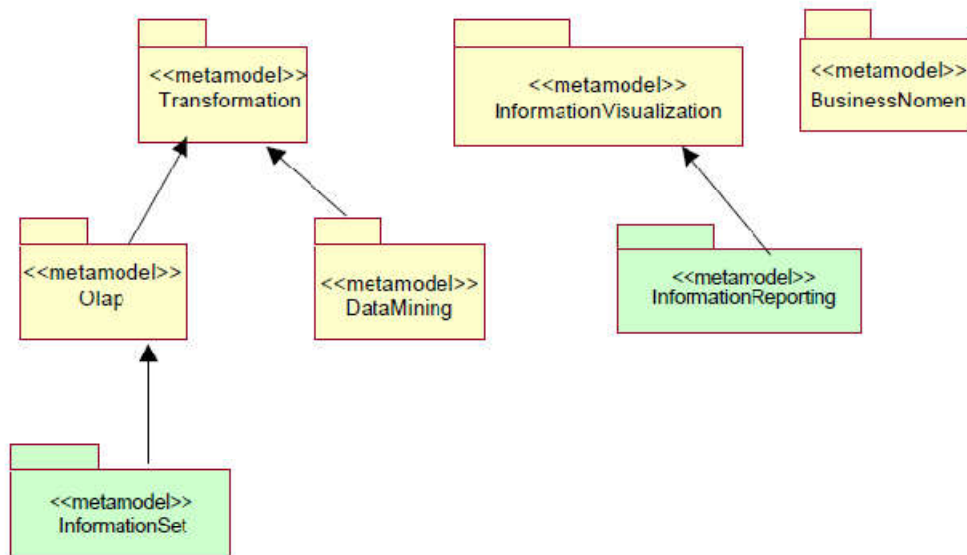
CWMX - Niveau supérieur UML CWM



CWMX – Fondation

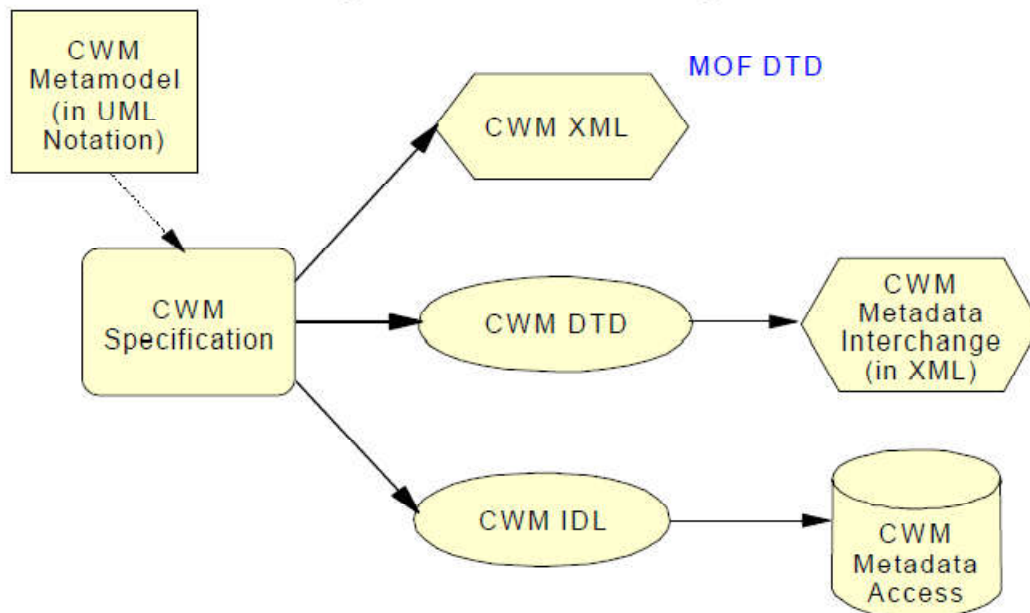


CWMX ressources



CWMX analyse

14. Les implantations du CWM



CWM Specification: CWM XML, CWM DTD, CWM IDL

En plus d'être un support pour la réalisation d'applications selon le MDA, le CWM est aussi un bon exemple de sa mise en œuvre. En effet, en définissant des méta-modèles indépendants pour les données et en permettant leurs projections vers des systèmes existants, le CWM suit la démarche du

MDA. De plus, il existe déjà plusieurs mises en œuvre. Les entreprises suivantes développent ou utilisent le CWM :

- Dimension EDI : leur but est de développer un dépôt de méta-données pour un système d'interrogation "intelligent". La plus grande partie de son développement est basé sur le CWM.
- Oracle Corporation [ORA] : les produits Oracle (Oracle Repository et Oracle8) permettent l'échange de données avec d'autres produits conformes au CWM.
- UBS [UBS] : UBS utilise le CWM pour migrer ses données vers de nouveaux environnements à base de composants, dans le domaine de la finance.
- Meta Integration Technology Inc [MIT] : le Meta Integration Model Bridge (MIMB) développé par cette société est certainement l'une des implantations du CWM les plus poussées.

–

15. Utilisateurs de CWM

CWM est visé à six catégories des utilisateurs :

1. constructeurs de la plateforme et de l'outil
2. Fournisseurs de service professionnel
3. Warehouse réalisateurs
4. Warehouse administrateurs
5. Utilisateurs
6. Gestionnaires de technologie de l'information

Ces utilisateurs participent à un ou plusieurs des quatre étapes suivantes dans le développement et l'utilisation des entrepôts CWM-basés de données :

1. Établissement : CWM mettant en application et se déployant, y compris un dépôt Service commun (suivant les indications du chiffre d'architecture de dépôt de méta-données de l'OMG de la préface).
2. Construction : exercer CWM pour définir une configuration d'entrepôt de données de ligne de base (établissant les voies d'accès d'échange entre les points d'émission et les cibles connus).
3. Operation : Fonctionnement de l'entrepôt CWM-basé de données.
4. Entretien : exercer CWM pour définir des changements de configuration d'entrepôt de données (aux changements de couverture aussi petits que l'ajout de plus d'éléments d'un type déjà de la configuration et aussi grands que la fusion ou le remplacement par une autre configuration).

Ce tableau montre comment CWM bénéficie des utilisateurs dans le développement et l'utilisation d'entrepôt de données :

Catégorie d'utilisateur	Problème ou besoin	Outils et dépôts	Comment CWM favorise mieux l'Utilisation d'entrepôt de données
Réalisateur d'entrepôt	Doit visualiser la source, cible, descriptions d'application (interfaces y compris).	Outils pour faciliter le développement avec la capacité de se rapporter à l'information dans le dépôt de méta-données.	CWM inclut des récipients pour la description aux niveaux de curation fine et secondaire.
Utilisateurs	Doit savoir <ul style="list-style-type: none"> • régénérer l'état d'entrées et de sorties des requêtes, • traçant entre les modèles pour le transfert des Modèles entre les outils, et • règles de transformation. 	Outils de requête et de présentation	CWM présente des modèles des méta-données à exploiter par des outils de requête et de présentation.
Gestionnaires de technologie de l'information	Doit avoir la visibilité dans l'état de déploiement d'entrepôt.	Outils de gestion de système et d'état	CWM présente des modèles des méta-données à exploiter par des outils de gestion de système et d'état.
	Doit souscrire aux normes	<ul style="list-style-type: none"> • CWM • Terrain 	CWM fournit une surface arrière commune pour des

Constructeurs de plateforme et d'outil d'entrepôt.	pour l'interconnexion d'intervendor.	communal de dépôt d'OMG Service • Outils pour modeler, développement, déploiement, et gestion de système	sous-ensembles que l'on peut brancher. C'est une notation globalement utilisable pour des protocoles d'échange de méta-données, qui permet la distribution flexible des services d'entreprise au-dessus d'une collection hétérogène de systèmes.
Fournisseurs de service professionnel	Doit accumuler et réutiliser des objets de l'enclenchement de service.	Les tiers et internes outils qui s'appliquent des méta-données de CWM à la base de données concrète catalogue et vice-versa.	Les méta-données réutilisables, éditables, et extensibles de CWM fournissent une base de capitaux qui établit la valeur. Cette base des objets réutilisables commence une boucle de feedback selfreinforcing avec des retours continuellement croissants (productivité améliorée d'enclenchement).
	Doit modifier la configuration : savoir ce qui et où modifier ; savoir la fermeture de dépendance.	Tiers ou internes outils pour contrôler l'édition de reconfiguration d'un modèle d'entrepôt. CWM expose l'information requis pour modifier un modèle.	La définition de contexte et les dispositifs selfdescribing de CWM sont employés pour isoler des rapports de dépendance.
Fournisseurs de service professionnel, administrateurs d'entrepôt	Doit intégrer les outils existants et les données qui adhèrent aux normes autres que	Outils basés sur la capacité de CWM d'incorporer des metamodels de legs, Web,	CWM fournit des sous-modèles supportant des détails d'information tenus dans une variété de différents formats, y compris XML, de SQL

	CWM dans une configuration d'entrepôt de données.	classe des propriétaires, et pratiques en matière et normes alternatives de définition de méta-données.	apparenté, et de formats conventionnels de fichier.
Administrateurs d'entrepôt	Doit établir et contrôler des expressions, des rapports, et des schémas multiples de base de données d'excédent de lignée.	Outils qui emploient les équipements intégrés de CWM pour définir le contenu, les rapports, et la lignée de schéma.	La conception de CWM est basée sur le besoin de contrôler une telle information aux niveaux multiples. Les modules de transformation et d'exécution d'entrepôt sont conçus pour permettre la navigation des méta-données corrélées avec des schémas.
	Doit ajouter, soustraire, la répartition, réapproprié, ou fusionner des ressources dans la configuration de déploiement.	Outils de gestion de système.	CWM se compose des modèles des méta-données qui aident à faire de tels changements et permettent à l'impact de ces changements d'être évalué.