

Objecteering

Version 6.0 SP2 : Notes de version

Table des matières

Table des matières.....	2
Modeler.....	3
CMS.....	4
C++ Developer (CxxDeveloper).....	4
Documentation.....	6
DOORS.....	6
Fortran Developer.....	6
Java Developer.....	6
MDA Modeler.....	7
RAMComponent.....	8
Requirements (RequirementsManager).....	8
SQL Designer.....	8

Modeler

- Les commandes "Enregistrer sous" et "copie" dans les diagrammes UML 1.4 sont maintenant disponibles en mode lecture seule.
- Le couper/coller des liens et des extrémités de liens fonctionne correctement.
- Les éléments abstraits sont maintenant affichés en italique.
- Des signaux peuvent maintenant être créés dans une interface.
- Les "Data Type" et les "Enumeration" peuvent maintenant être démasqués dans les interfaces.
- Des "UseCases" peuvent maintenant être créés dans les classes.
- Le déploiement du MDAC MDAModeler n'affiche plus d'erreurs.
- Amélioration du fonctionnement des RAMCs.
- Le menu "Fichier/ouvrir" ouvre directement le dossier "work".
- "Interface realization" et "template binding" sont maintenant représentés différemment.
- Les "Enumerations" peuvent maintenant être créées dans les diagrammes de classes.
- Il est maintenant possible de démasquer un lien par un glisser/déposer.
- Objecteering ne plante plus sur un Ctrl-D dans les diagrammes UML 1.4.
- Il est maintenant possible de sélectionner un rôle d'association dans l'explorateur depuis un diagramme de classe.
- La commande "Unmask/All contents" sur package démasque maintenant tout les éléments.
- Les accents sont maintenant correctement gérés dans la description des RAMCs.
- Les interfaces sont maintenant abstraites par défaut.
- Objecteering peut être utilisé en ligne de commande.
- Les valeurs des paramètres des MDACs sont conservés lors de la mise à jour des MDACs.
- La sélection d'un rôle dans un diagramme rafraîchit la fenêtre auxiliaire.
- Les paramètres des Tagged values sont maintenant affichés dans les diagrammes.
- Le champs "Type Constraint" de la boîte d'édition des paramètres de retour a été renommé "Type constraint / String size".
- Correction d'une erreur d'Audit dans les diagrammes de séquence.
- Les noms des Dataflows sont maintenant affichés dans les diagrammes.
- La case est maintenant prise en compte pour les valeurs initiales des Attributs dans boîte de propriété Java.
- Les diagrammes de Collaboration UML 1.4 ne sont plus migrés en diagrammes UML 2.0, il est maintenant possible de créer les deux sortes de diagrammes : Collaboration UML 1.4 et Composite structure UML 2.0.
- L'entrée du menu d'aide correspond maintenant aux standards IHM.
- Les méthodes "destructor" peuvent maintenant être déclarées abstraites.
- Les cycles entre DataFlows ne sont plus signalés par l'Audit.
- Les Séquence Messages ayant des Dépendances avec des liens <<throw>> sont maintenant correctement migrés.
- Correction d'un plantage survenant lors la connexion d'un projet Requirements à un référentiel GVC non Requirements.
- Amélioration dans la suppression des Séquence Messages.
- Correction d'un plantage survenant lors d'une tentative de création d'un lien de traçabilité entre un élément de modèle et un Requirement en mode Lecture Seule.
- Le bundle WinMerge a été mis à jour.
- Les notes peuvent maintenant être créées/démasquées directement sur un diagramme.

- Correction d'un plantage survenant lors de la création de Class Associations dans les diagrammes de classe.
- Correction d'un plantage survenant lors du démasquage de Template Bindings entre 2 Usecases dans les diagrammes de classe.
- Les produits de génération ne peuvent plus être édités plus d'une fois simultanément.
- Il est maintenant possible de copier/coller les stéréotypes.
- La fenêtre auxiliaire est désormais rafraîchie au deployment d'un MDAC.
- Il est maintenant possible d'ajouter des Notes, Tagged Values, Stereotypes, et Contraintes sur les Enumeration Literals
- Les Tagged values sont correctement gérées via les onglets des MDACs.
- Les RAMCs sont correctement gérés par la fonctionnalité Diff Merge.
- Correction d'un plantage survenant lors du drag'n'drop d'une Classe Template vers sa propre boîte de dialogue.
- Le nom des "Template Parameters" est désormais correctement affiché lors du drag'n'drop vers un type d'attribut.
- Les produits de génération sont désormais correctement propagés.
- Le module "CxxModule" n'est plus automatiquement remplacé par le MDAC CxxDeveloper lors de la migration d'une base Objectceering 5.3.0.
- Correction d'une erreur apparaissant lors du check-in avec récupération du code Java.
- Amélioration de la gestion mémoire lors du packaging de MDACs.
- Les couleurs sont désormais correctement sauvegardées dans les diagrammes d'état.
- Les diagrammes sont de nouveau présents dans la documentation générée.
- Les Required et Provided Interfaces ne peuvent désormais être créées qu'entre des Component/Class/Part/Instance et des Part/Instance.
- Les erreurs survenant Durant le merge automatique sont affichées dans un rapport final.
- Correction d'un bug concernant les couleurs des attributs dans les diagrammes.
- Les bulles d'aide des icônes de la barre d'outil sont affichées.
- Mise à jour des volumes documentaires.

CMS

- "PVCSDimensionsModule" a été renommé "Dimensions".
- Correction d'un bug pouvant survenir lors de l'utilisation des imports non hiérarchiques.
- Les RAMCs sont maintenant automatiquement déployés si nécessaire, à l'aide du fichier de configuration "RamcList.conf".
- Le paramètre 'Update generation work products before check-in' est maintenant pris en compte par CxxDeveloper.
- Les fichiers "*.read", "*.write" et "multiuser.lock" sont maintenant situés dans le répertoire \$MU/Locks/users/. Ceci afin d'améliorer les performances de verrouillage du référentiel depuis un réseau distant.
- Le MDAC Dimensions supporte désormais Merant Dimensions 8.0 et Serena Dimensions 9.1
- Le MDAC CMSynergy supporte désormais CMSynergy 6.3 et 6.4 (Le répertoire 'Locks' n'est plus géré en version. Seuls les projets CM Synergy en mode « shared » sont supportés. Les projets Personnel/Insulated ne le sont plus. Les référentiels existants doivent être recréés par l'administrateur CM Synergy).

C++ Developer (CxxDeveloper)

- La création automatique des constructeurs, destructeurs et opérations est maintenant activée quand une classe est créée dans un diagramme.
- Les "Enumerations" et "Enumerations literals" sont maintenant gérés par la génération doxygen.
- Les cardinalités des paramètres sont maintenant prises en compte.
- Ajout d'une nouvelle option : " Use description as doxygen"
- Les méthodes abstraites sont maintenant automatiquement déclarées virtuelles.
- Les cardinalités des paramètres de retour sont maintenant prises en compte.
- Remplir le champs "throws" d'une opération avec "(" l'empêche de lancer une exception.
- Les classes internes sont maintenant correctement déclarés en avance.
- Les directives d'inclusions sont maintenant correctement générées si une classe déclare un constructeur par recopie.
- Les directives d'inclusions sont maintenant correctement générés pour les interfaces génériques.
- Le code de base des constructeurs est maintenant généré pour les constructeurs avec paramètres.
- Le champs "argument" de la boite d'édition des associations est maintenant pris en compte.
- Toutes les exceptions lancées sont maintenant générées.
- La déclaration des méthodes virtuelles pures lançant des exceptions est maintenant correctement générée.
- La tagged value `{virtual}` n'est plus prise en compte sur les methodes 'cannot be specialized'.
- Correction dans la génération des ".hpp" des packages pour les types de données qui instancient des classes génériques.
- Le champs "Arguments" de la boite de propriété C++ est maintenant correctement pris en compte.
- Il est maintenant possible d'éditer la documentation sur les littéraux d'énumération.
- Les notes Doxygen sont maintenant générées dans le fichier .Hpp avec des marqueurs.
- Le code des fichiers d'en-tête est maintenant correctement remonté après une édition externe.
- La déclaration manuelle d'une opération fonctionne correctement.
- Les méthodes virtuelles pures ne sont générées dans un fichier source que si l'opération contient une note 'Cxx.Code'.
- Les Class templates ayant un attribut 'static const' sont correctement générées.
- Il n'y a plus d' erreur J lorsqu'une exception est lancée vers une classe tagguée `{NoCode}`.
- Le type de pointeur d'un qualifieur est pris en compte dans la génération des map key.
- Les attributs taggués `{NoCode}` ne sont plus générés.
- Amélioration générale de la conversion des anciennes tags C++.
- Les boutons '?' de la fenêtre d'édition de projet fonctionnent correctement.
- CxxDeveloper est maintenant complètement couplé avec les MDACs de GVC.
- Amélioration générale de la gestion des Projets.
- La génération est maintenant verbeuse.
- Le mot clé "Inline" n'est plus ajouté à la déclaration des méthodes abstraites.
- Les "ElementImport" des Enumérations sont maintenant générés.
- Ajout du paramètre "Multiple return parameter by collection", permettant de générer des collections pour les paramètres de retour.
- Les tagged values `{Virtual}` sont supprimées si les opérations abstraites sont changées en "Cannot be specialized".
- Les Attributs, les Paramètres et les Associations sont maintenant initialisés avec la note Cxx.value si elle est présente. Autrement la valeur UML est utilisée.
- Ajout d'un paramètre "Automatic guessing strategy", permettant de choisir si la décoration doit être automatique ou déclenchée manuellement.

- Ajout d'un paramètre "Directory for personal ACT files", permettant à l'utilisateur d'employer ses propres fichiers ACT.
- Introduction du concept de "Smart Generation". Une génération qui saute les éléments qui n'ont pas besoin d'être régénérés, à l'opposé de la "Forced generation", qui régénère le projet entier.
- Ajout d'un paramètre "Default behavior for file retrieving", permettant de choisir le comportement de la génération lorsqu'un fichier peut être mis à jour. Les différents choix sont : 'ask', 'retrieve all files' ou 'force the regeneration'.
- Il est désormais possible d'ajouter un fichiers externe ou un groupe de fichiers externes à compiler avec l'application.
- Les anciennes tagged values C++ {*} sont correctement converties.
- Correction d'une erreur J apparaissant au lancement de la commande "Convert old C++ tags" sur les modèles migrés contenant des stéréotypes en read only.
- Les associations Static Const avec un type de container "list" sont désormais correctement générées.
- Correction d'une erreur J apparaissant au lancement d'un check-in sur les projets Requirements.
- Les associations entre Classes et Datatypes sont maintenant correctement générées.
- Correction d'une erreur J apparaissant à la génération de modèle contenant des tagged values {}Cxx.Name ou {}Cxx.Container vides.
- Les accesseurs "Get" sont désormais générés comme "Const".

Documentation

- Les profiles, les modèles et le code J sont maintenant visibles et modifiables afin de pouvoir les importer et les redéfinir dans un nouveau projet MDA.

DOORS

- Il est maintenant possible d'importer des modules DOORS contenant des caractères utf-8.
- Amélioration dans la gestion des droits utilisateurs durant l'opérations "export".
- Amélioration du système des opérations d'import.
- Les "Objects below" DOORS nouvellement créés sont maintenant correctement importés dans Objecteering.

Fortran Developer

- Les modèles migrés depuis la 5.3.0 sont correctement générés.

Java Developer

- "Enumerations" et "enumerations literals" sont maintenant gérées par le moteur du "reverse" et la génération de javadoc.
- Amélioration du système de génération des imports.
- Les classes stéréotypées " <<interface>> " sont maintenant générées comme des interfaces.
- Correction d'un bug concernant le reverse de classes directement sous le package racine.

- Les javadoc sont maintenant correctement récupérées sur les littéraux d'énumération en mode Model Driven.
- Correction d'un bug concernant le reverse des énumérations contenant des méthodes et des attributs.
- Correction dans la génération des méthodes d'une interface lançant des exceptions.
- La tagged value {}JavaBind est maintenant prise en compte pour les liens de généralisation entre classes génériques.
- La tagged value {} javaExtern est maintenant prise en compte pour les types de données dans les classes.
- La tagged value {}nocode est maintenant prise en compte.
- Les produits de génération sont maintenant correctement propagés.
- Les commandes 'Create the methods to be implemented/Redefined' peuvent être lancées depuis les RAMCs.
- Objecteering génère maintenant des fichiers ANT au lieu de fichiers makefiles.
- Il est maintenant possible de choisir la visibilité "undefined" avec l'onglet de propriété Java.
- Amélioration générale de la génération des tagged values "@param" et "@return".
- La visibilité des Enumérations est maintenant prise en compte.
- Corrections sur la génération de la tagged value {}JavaTypeExpr.
- Ajout d'une tagged value {}JavaNocode inhibant la génération des éléments.
- Ajout d'un champs "Key Type" dans l'onglet de propriété Java, disponible pour les Associations, Attributs et Paramètres ayant une cardinalité multiple.
- Correction sur la génération des Classes contenant des Enumérations implémentant des Interfaces.
- Les Interfaces contenus dans les Classes sont maintenant générées.
- Amélioration générale du reverse
- Ajout d'une annotation "objingid" sur les classes, interfaces, énumérations, attributs, associations et opérations, permettant au Round Trip de retrouver les éléments dans le projet Objecteering lorsqu'ils sont reversés.
- Des boutons 'Generate java code', 'Generate Ant file', 'Build Ant file', 'Create a Jar file' et 'Execute Ant target' ont été ajoutés à l'onglet Java des JarFiles.
- Amélioration générale du reverse des generics.
- Les template parameters et énumérations des interfaces sont reversés.
- Les énumérations littérales sont maintenant correctement reversées et générées.
- Les annotations sont reversées pour les constructeurs.
- Les JavaAnnotations sont maintenant correctement reversés.
- Les annotations @return inutiles ne sont plus générées sur les setters.
- Les marqueurs du mode Model Driven ne sont plus générés dans le mode Round Trip.
- Un message d'erreur compréhensif est affiché quand le reverse essaye de créer des éléments sous des éléments Read-Only.
- L'option 'Automatically create diagrams on initial reverse' fonctionne maintenant correctement.
- Le bouton 'Run an Applet' fonctionne maintenant correctement.

MDA Modeler

- La sélection d'un dossier pour exporter un pattern est maintenant autorisée.
- Correction d'une erreur J quand un pattern est appliqué avec MDARuntime.
- La mise à jour du "framework" est maintenant possible.
- Ajout d'un label DocBook pour générer un champ RTF pour l'inclusion de fichier. Ce label génère un hyperlien pour les formats HTML et PDF.
- La sélection de modèles est maintenant possible.

- La génération des fichiers XML (DocBook) est maintenant autorisée.
- Le dossier des ressources est maintenant créé avant le "packaging".
- Ajout des bulles d'aide pour les icônes dans l'onglet de propriétés MDA.
- Ajout d'un nouveau bouton pour créer d'anciens modèles de documentation.

RAMComponent

- La command "Remove model component" fonctionne correctement.
- L'extension ".ramc" est correctement gérée dans les boites de sélection de fichiers.

Requirements (RequirementsManager)

- Correction d'un problème bloquant la modification des objets d'exigences dans la documentation générée.
- Mise à jour de la barre d'outil "Requirements" de la documentation générée.
- Correction d'un problème de présentation dans la documentation générée.
- L'extension ".XML" est maintenant positionnée par défaut lors de la demande du chemin de génération du fichier XML pour la documentation générée.
- Ajout d'un package Uno (Open Office) permettant l'export de documents Open Office vers Objecteering.

SQL Designer

- Les diagrammes générés avec le modèle physique sont en version UML2.
- Les diagrammes accompagnant le "First Steps" sont en version UML2.
- Corrections dans les types prédéfinis pour Oracle.
- Les bulles d'aides sont disponibles pour la barre d'outil de SQLDesigner.
- Les modifications de parameters de la commande 'Run SQL' sont désormais prises en compte.
- Le champ 'SQL type' sur les attributes permet désormais de choisir entre tous les types accessibles.
- Le champ 'SQL type constraint' peut désormais être utilisé pour tous les types (UML, SQL Server, Oracle...)
- Les parameters IN OUT Oracle d'une procedure stockée sont désormais correctement générés.
- La valeur par défaut des CHAR et VARCHAR Sybase et SQLServer n'est plus renseignée automatiquement.
- Il est désormais possible de générer du code SQL lorsque « Max » est sélectionné pour un paramètre d'opération.