ENTERPRISE ARCHITECT

Série de Guides d'Utilisateur

Ingénierie de l'information

Author: Sparx Systems
Date: 23/11/2023
Version: 16.1

CRÉÉ AVEC ENTERPRISE ARCHITECT

Table des Matières

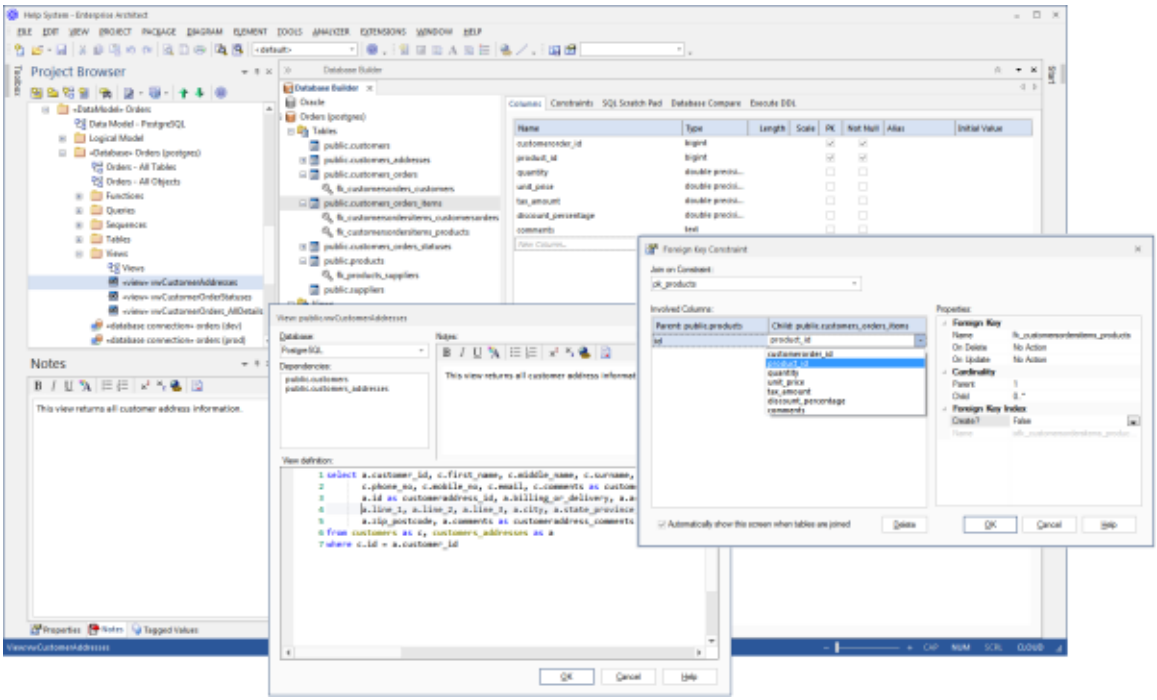
Ingénierie de l'information 4
Exemple Diagramme 5
Travailler avec des types Modèle de données 6
Modèle de données conceptuelles 7
Diagrammes Entité-Relation (ERD) 8
Modèle de données logiques 12
Modèles de données physiques 13
Transformation DDL 15
Plus d'information 20
Systèmes de gestion de base de données pris en charge 21
Générer un langage de définition de base de données (DDL) 22
Générer du DDL pour les objets 23
Modifier Gabarits DDL 27
Syntaxe DDL Gabarit 29
Gabarits DDL 30
Gabarits de base pour la génération DDL 31
Gabarits de base pour la génération Alter DDL 35
Macros DDL 36
Macros de champs d'éléments 37
Macros de champs de colonnes 40
Macros de champs de contraintes 41
Macros de fonctions DDL 43
Macros de propriétés DDL 48
Options DDL dans Gabarits 54
Limites du DDL 57
Importer le schéma de base de données 59
Création et gestion de modèles de données 63
Paramètres Modélisation des données 64
Notations Modélisation des données 65
Nom DDL Gabarits 70
Types de données 72
Mapper les types de données entre les produits SGBD 73
Conversion de produit SGBD pour un Paquetage 75
Conversion Type de données pour un Tableau 76
Types de données de base de données 77
Types de données MySQL 79
Types de données Oracle 80
Gérer les options du SGBD 81
Objets de base de données 83
Tableaux de base de données 84
Créer un Tableau de base de données 86
Colonnes Tableau de base de données 88
Créer des colonnes Tableau de base de données 89
Supprimer les colonnes Tableau de la base de données 92
Réorganiser les colonnes Tableau de la base de données 93
Utilisation Propriétés Tableau de base de données 94
Définir le Type de base de données 95

Définir le propriétaire/schéma de la base de données Tableau 96
Définir les options MySQL 97
Définir Propriétés Tableau de base de données Oracle 98
Contraintes/index de la base de données Tableau 99
Primary Keys 102
Primary Keys non clusterisées 105
Index de base de données 106
Contraintes uniques 109
Foreign Keys 110
Vérifier les contraintes 114
Tableau Déclencheurs 116
Vues de base de données 118
Procédures de base de données 120
Fonctions de base de données 123
Séquences de base de données 125
Requêtes SQL de base de données 128
Créer des conteneurs d'opérations 130
Paquetages Oracle 132
Connexions à la base de données 133
Le constructeur de base de données 136
Contraintes/Index 138
Contraintes/index de la base de données Tableau 139
Primary Keys 142
Index de base de données 145
Contraintes uniques 148
Foreign Keys 149
Vérifier les contraintes 153
Tableau Déclencheurs 155
Colonnes 157
Créer des colonnes Tableau de base de données 158
Supprimer les colonnes Tableau de la base de données 161
Réorganiser les colonnes Tableau de la base de données 162
Travailler dans le générateur de base de données 163
Ouverture du générateur de base de données 167
Bloc-notes SQL 169
Exécuter le DDL 171
Comparaison de base de données 174
Créer un Diagramme Modèle de données 181
Exemple Diagramme Modèle données 183
Créer un Modèle de Données à partir d'un Modèle Motif 184
Démarrage 186

Ingénierie de l'information

Concevoir, créer et gérer des modèles de données conceptuels, logiques et physiques

La puissance du développement de systèmes basés sur des modèles réside dans la capacité de visualiser, d'analyser et de concevoir tous les aspects d'un système. Être capable d'afficher et de gérer des informations et des données aux côtés d'autres modèles d'un système offre une grande clarté et réduit le risque d'erreur. Enterprise Architect prend support de manière étendue la discipline modélisation de données, allant de la représentation d'informations dans un modèle conceptuel jusqu'à la génération d'objets de base de données. Que vous génériez des objets de base de données à partir du modèle UML ou que vous procédiez à l'ingénierie inverse d'un SGBD existant dans un modèle d'analyse, les fonctionnalités de l'outil vous permettront d'économiser du temps et des ressources de projet précieuses.



Cette illustration montre l'interface du générateur de base de données, y compris la génération DDL et la dialogue Foreign Key .

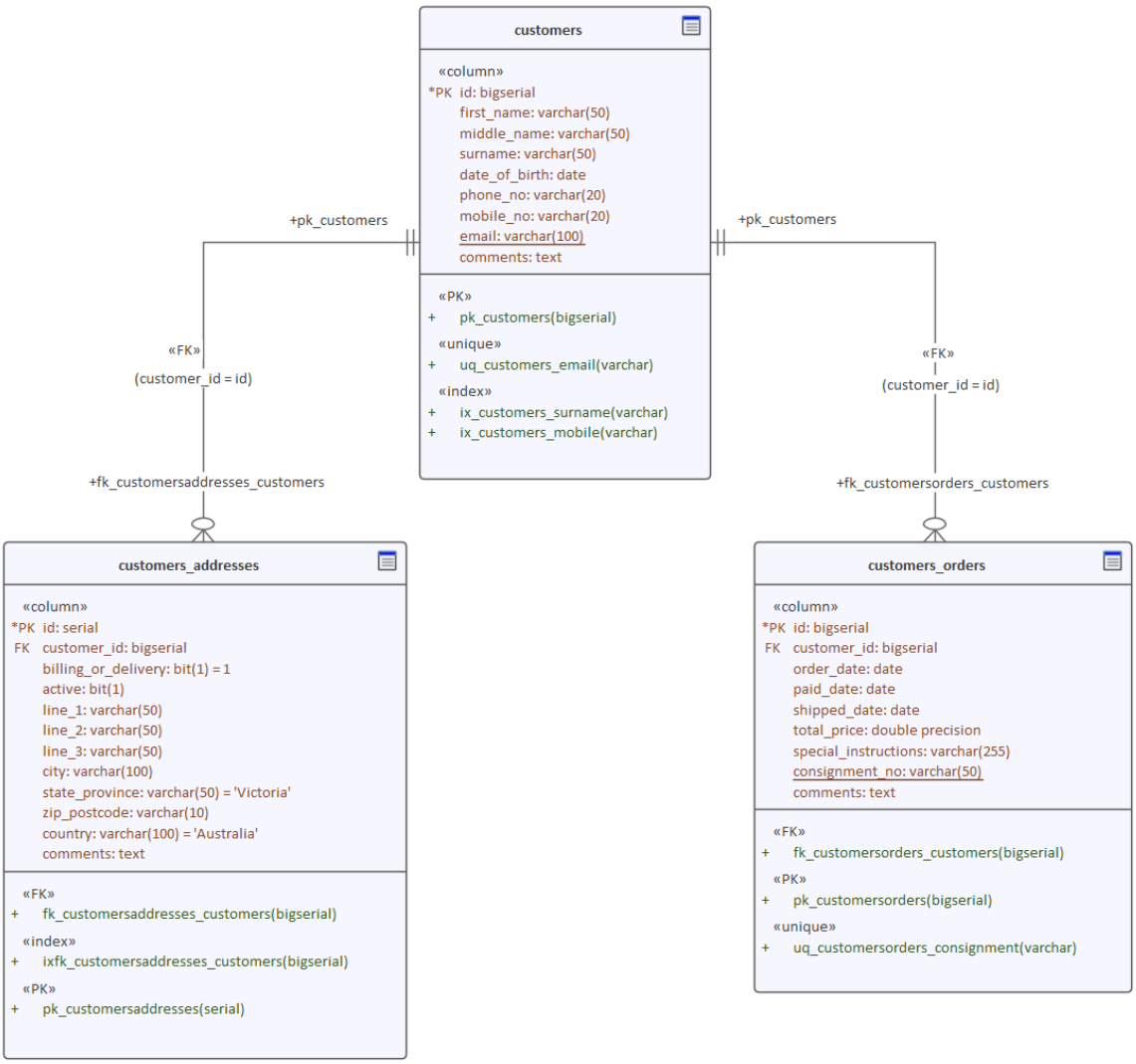
Enterprise Architect supporte la modélisation des informations au niveau conceptuel, logique et physique. Grâce à un certain nombre de fonctionnalités standards, ces modèles peuvent être interconnectés, assurant ainsi la traçabilité. Les modèles logiques et physiques peuvent également être générés automatiquement à l'aide d'un moteur de transformation entièrement personnalisable. Les systèmes existants peuvent être importés, analysés et comparés à l'aide de la facilité d'ingénierie inverse pratique.

Dans cette rubrique, vous apprendrez à utiliser l'ensemble d'outils riches fonctionnalité , notamment Database Builder, pour concevoir, créer, gérer et visualiser des données, y compris l'ingénierie inverse et directe de modèles de données vers une base de données en direct.

L'outil Database Builder peut être utilisé pour créer et maintenir des modèles de données physiques et peut se connecter à un SGBD en cours d'exécution, vous pouvez donc importer, générer, comparer et modifier une base de données en direct.

Exemple Diagramme

En utilisant les fonctionnalités d'ingénierie de base de données d' Enterprise Architect vous pouvez créer des modèles riches des objets qui composent un modèle de données à n'importe quel niveau d'abstraction, du conceptuel au logique en passant par le physique. Ces modèles sont créés en ajoutant tableaux et d'autres objets de base de données à partir de la boîte à outils ou par rétro-ingénierie et base de données existante dans un modèle à partir d'une gamme de SGBDR. Un diagramme de base de données peut contenir Tableaux , des vues, des procédures, des séquences et des fonctions. Les colonnes Tableau sont annotées car Foreign Keys sont modélisées à l'aide de relations d'association spécialisées. Dans cet exemple, l'utilisateur a créé un modèle de données physique simple des clients, de leurs adresses et commandes.



Modèle de données physiques montrant Tableaux avec des colonnes et Foreign Keys .

Travailler avec des types Modèle de données

Enterprise Architect fournit un certain nombre de fonctionnalités pour faciliter le processus de création de modèles d'informations, notamment la possibilité de développer des modèles conceptuels, logiques et physiques et de pouvoir retracer les concepts sous-jacents entre les modèles. Les modèles physiques peuvent être développés pour une large gamme de systèmes de bases de données, et l'ingénierie directe et inverse permet à ces modèles d'être synchronisés avec des bases de données en direct.

Modèles de données

Type	Description
Modèles de données conceptuels	Les modèles de données conceptuels, également appelés modèles de domaine, établissent les concepts de base et la sémantique d'un domaine donné et aident à les communiquer à un large public de parties prenantes. Les modèles conceptuels servent également de vocabulaire commun lors des étapes d'analyse d'un projet ; ils peuvent être créés dans Enterprise Architect à l'aide de modèles Entité-Relation ou Classe UML .
Modèles de données logiques	Les modèles de données logiques ajoutent des détails supplémentaires aux éléments du modèle conceptuel et affinent la structure du domaine ; ils peuvent être définis à l'aide de modèles Entité-Relation ou Classe UML . L'un des avantages d'un modèle de données logique est qu'il fournit une base sur laquelle baser le modèle physique et la mise en œuvre ultérieure de la base de données. modélisation entité-relation est une méthode modélisation de base de données abstraite et conceptuelle, utilisée pour produire un schéma ou un modèle de données sémantique, par exemple, d'une base de données relationnelle et de ses exigences, visualisée dans Diagrammes entité-relation (ERD). Les ERD vous aident à créer des modèles de données conceptuels jusqu'à générer un langage de définition de données (DDL) pour le SGBD cible. Un modèle logique peut être transformé en modèle de données physique à l'aide d'une transformation DDL.
Modèles de données physiques	Les modèles de données physiques dans Enterprise Architect vous aident à visualiser la structure de votre base de données et à dériver automatiquement le schéma de base de données correspondant ; vous utilisez le profil UML d' Enterprise Architect pour Modélisation des données spécifiquement à cette fin. Le profil fournit des extensions utiles de la norme UML qui mappent les concepts de bases de données de Tableaux et de relations sur les concepts UML de classes et d'associations ; vous pouvez également modéliser les colonnes de la base de données, les clés, les contraintes, les index, déclencheurs , l'intégrité référentielle et d'autres fonctionnalités de base de données relationnelles. Étant donné Enterprise Architect vous aide à visualiser chaque type de modèle de données dans le même référentiel, vous pouvez facilement gérer les dépendances entre chaque niveau d'abstraction pour maximiser la traçabilité et vérifier l'intégralité de la mise en œuvre du système.

Modèle de données conceptuelles

Un modèle de données conceptuel est la forme la plus abstraite de modèle de données. Il est utile pour communiquer des idées à un large éventail de parties prenantes en raison de sa simplicité. Par conséquent, les informations spécifiques à la plateforme, telles que les types de données, les index et les clés, sont omises d'un modèle de données conceptuel. D'autres détails d'implémentation, tels que les procédures et les définitions d'interface, sont également exclus.

Il s'agit d'un exemple de modèle de données conceptuel, rendu à l'aide de deux des notations prises en charge par Enterprise Architect .

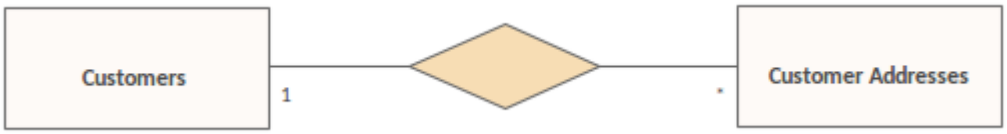


diagramme de relation d'entité montrant une relation un-à-plusieurs

En utilisant la notation Entité-Relation (ER), nous représentons les concepts de données « Clients » et « Adresses des clients » comme des entités avec une relation un-à-plusieurs entre elles. Nous pouvons représenter exactement les mêmes informations sémantiques en utilisant les classes et associations UML .

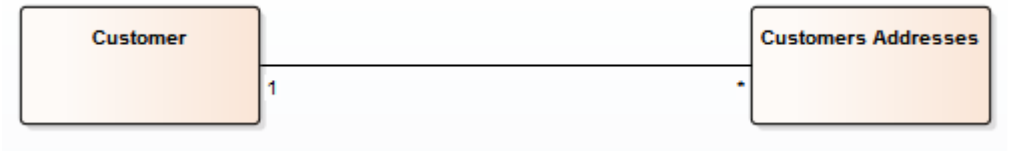


diagramme Unified Modeling Language montrant la même relation un-à-plusieurs

Le fait que vous utilisiez la notation UML ou ER pour représenter les concepts de données dans votre projet dépend de l'expérience et des préférences des parties prenantes impliquées. La structure détaillée des concepts de données illustrés dans un modèle de données conceptuel est définie par le modèle de données logique.

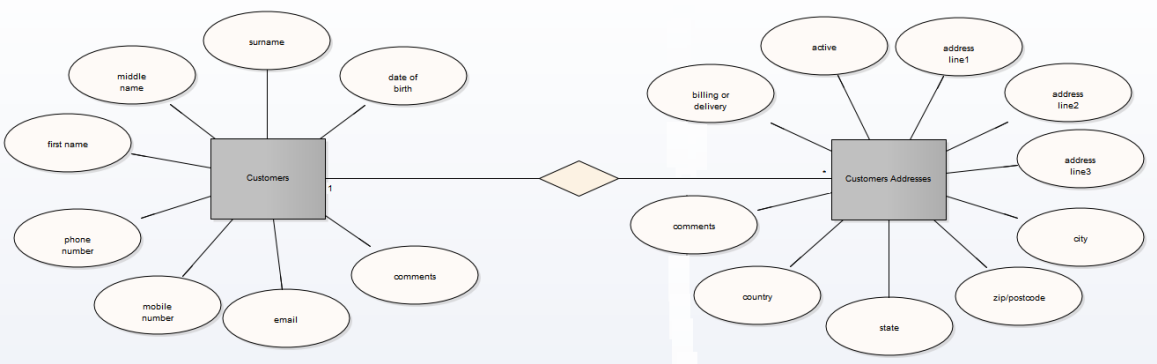
Diagrammes Entité-Relation (ERD)

D'après Wikipédia en ligne :

Un modèle entité-relation (ERM) est une représentation abstraite et conceptuelle des données. modélisation entité-relation est une méthode modélisation de base de données, utilisée pour produire un type de schéma conceptuel ou de modèle de données sémantique d'un système, souvent une base de données relationnelle, et ses exigences de manière descendante. Diagrammes créés par ce processus sont appelés Diagrammes entité-relation, Diagrammes ER ou ERD.

Diagrammes de relation d'entité dans Enterprise Architect

diagrammes de relations entre entités dans Enterprise Architect sont basés sur les blocs de construction ERD de Chen : les entités (tableaux) sont représentées sous forme de rectangles, les attributs (colonnes) sont représentés sous forme d'ellipses (jointes à leur entité) et les relations entre les entités sont représentées sous forme de connecteurs en forme de losange.



La technologie ERD d' Enterprise Architect vous assiste à chaque étape, depuis la création de modèles de données conceptuels jusqu'à la génération du langage de définition de données (DDL) pour le SGBD cible.

Transformations ERD et ERD

Enterprise Architect vous permet de développer diagrammes Entité-Relation rapidement et simplement, grâce à l'utilisation d'une MDG Technologie intégrée au programme d'installation Enterprise Architect .

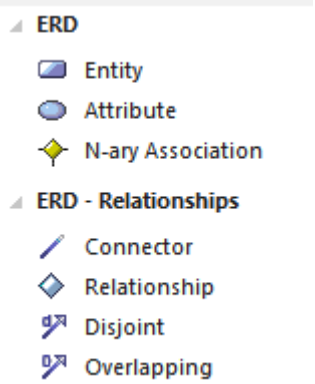
Les diagramme Entité-Relation facilités se présentent sous la forme de :

- Un type diagramme Entité-Relation, accessible via la dialogue ' Nouveau Diagramme '
- Une page Diagramme Entité-Relation dans la boîte à outils Diagramme
- Élément de relation d'entité et entrées de relation dans le menu « Raccourci de la boîte à outils » et Quick Linker

Enterprise Architect fournit également gabarits de transformation pour transformer diagrammes Entité-Relation en diagrammes Modélisation de Données, et vice versa.

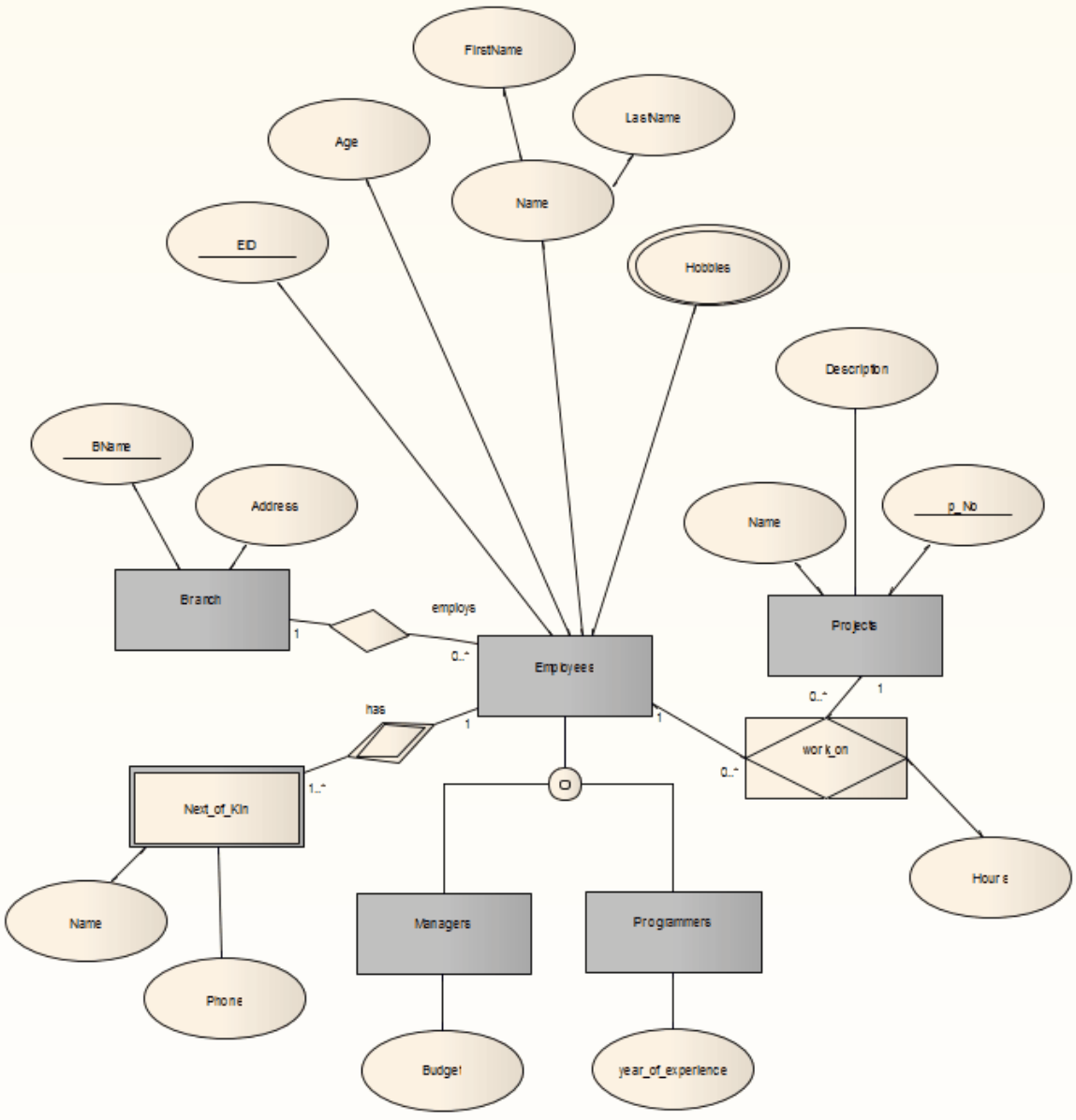
Page de la boîte à outils du Diagramme de relation d'entité

Vous pouvez accéder à la page ' Diagramme Entité-Relation' de la Boîte Diagramme en spécifiant ' Diagrammes Entité-Relation' dans la boîte de dialogue ' Trouvez Item de Boîte à Outils dialogue de la Boîte à Outils.



- L'entité est un object ou un concept identifiable de manière unique ; la propriété 'Multiplicity' dans les définitions SourceRole et TargetRole pour le connecteur Relation peut être utilisée pour définir la cardinalité d'une entité qui participe à cette relation
- L'attribut est une propriété d'une entité ou d'un type de relation
- L'association N-aire représente des relations unaires (plusieurs à plusieurs récursives) ou ternaires et peut également être utilisée pour représenter des relations qui ont des attributs entre les entités ; l'élément d'association N-aire doit toujours être à l'extrémité cible d'un connecteur
- Le connecteur est un connecteur entre une entité et un attribut, et entre deux Attributes
- La relation est un connecteur en forme de losange, représentant l'association significative entre les entités
- Disjoint et Overlapping représentent les relations entre l'entité de super-classe et l'entité de sous-classe

Un diagramme Entité-Relation typique



Valeur Étiquetés

Certains composants du diagramme Entité-Relation peuvent être modifiés par Valeur Étiquetés , comme indiqué :

Composant	Valeur Étiquetée / Notes
Entité	estEntitéFaible Notes : Si vrai, cette entité est une entité faible.
Attribut	Type d'attribut Notes : Il existe quatre options valides : "normal", " primary key ", "à valeurs multiples" et "dérivé".
Attribut	CommonDataType Notes : Définit le type de données commun pour chaque attribut.

Attribut	dbmsDataType Notes : Définit le type de données SGBD personnalisé pour chaque attribut. Cette option n'est disponible que lorsque la balise *commonDataType* est définie sur « na ». Vous devez d'abord définir le type personnalisé via l'option du ruban « Paramètres > Données de référence > Paramètres > Types de données de base de données ».
Association N-aire	estRécursif Notes : Si c'est vrai, l'association N-aire représente la relation récursive plusieurs-à-plusieurs. Pour les relations récursives un-à-plusieurs et un-à-un, nous suggérons d'utiliser le connecteur Relation normal. Parfois, vous souhaiterez peut-être limiter l'étirement des connecteurs de relation en forme de losange ; choisissez simplement un connecteur de relation, cliquez-droit pour afficher le menu contextuel et sélectionnez l'option « Bend Line at Cursor ».
Relation	est faible Notes : Si vrai, la relation est une relation faible.
Séparer Chevauchement	Participation Notes : Il existe deux options valides, « partiel » et « total ».

Notes

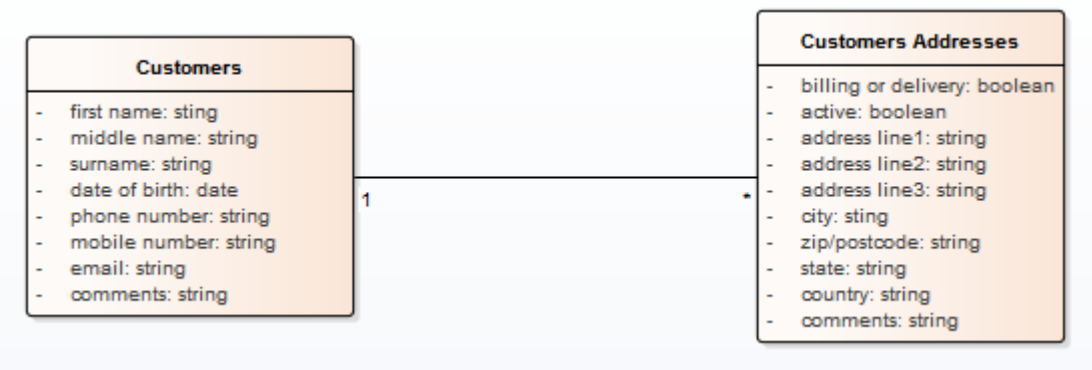
- diagrammes de relations entre entités sont pris en charge dans les éditions Corporate , Unified et Ultimate d' Enterprise Architect

Modèle de données logiques

Les modèles de données logiques aident à définir la structure détaillée des éléments de données dans un système et les relations entre les éléments de données. Ils affinent les éléments de données introduits par un modèle de données conceptuel et constituent la base du modèle de données physique. Dans Enterprise Architect , un modèle de données logique est généralement représenté à l'aide de la notation de classe UML .

Exemple

Ce diagramme est un exemple simple de modèle de données logique. Le Modèle Logique ajoute des détails au Modèle Conceptuel mais sans aller jusqu'au niveau de spécification du système de gestion de base de données qui sera utilisé.



Modèle de Données Conceptuelles avec tableaux modélisation Clients et leurs Adresses.

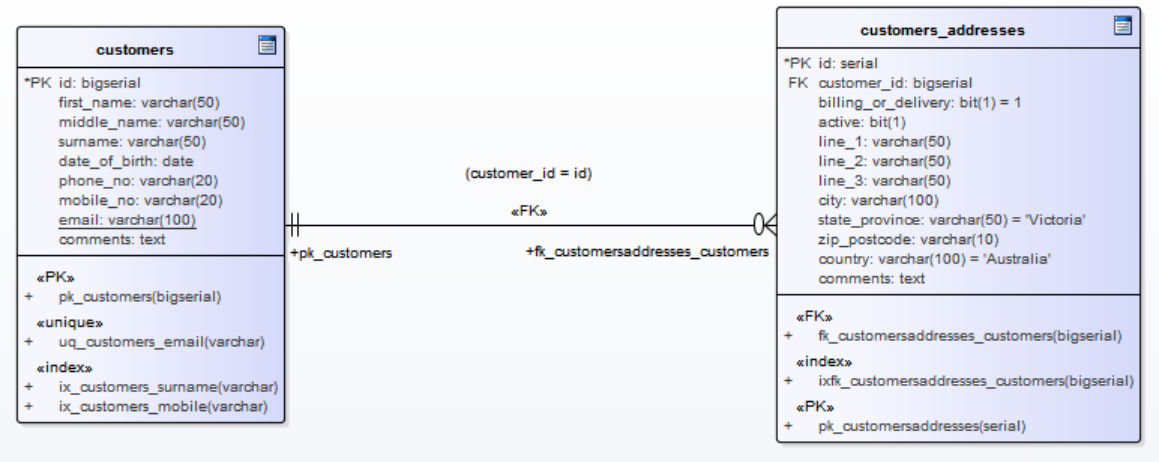
Note que les éléments de données « Clients » et « Adresses des clients » contiennent des attributs UML ; les noms et les types de données génériques restent indépendants de la plate-forme. Les types de données spécifiques à la plate-forme et autres métadonnées liées à une implémentation spécifique du SGBD sont définis par le modèle de données physique.

Modèles de données physiques

Un Modèle de données physiques représente visuellement la structure des données telle que mise en œuvre par un schéma de base de données relationnelle. En plus de fournir une abstraction visuelle de la structure de la base de données, un avantage important de la définition d'un Modèle de données physiques est que vous pouvez automatiquement dériver le schéma de la base de données à partir du modèle. Cela est possible grâce à la richesse des métadonnées capturées par un Modèle de données physiques et à son mappage étroit avec des aspects du schéma de base de données, tels que Tableaux de base de données, les colonnes, Primary Keys et Foreign Keys .

Exemple Modèle de données

Cet exemple montre un Modèle de données physiques qui pourrait être utilisé pour générer automatiquement un schéma de base de données. Chaque Tableau est représenté par une classe UML ; Les colonnes Tableau , Primary Keys et Foreign Keys sont modélisées à l'aide d'attributs et d'opérations UML . Ce modèle démontre l'utilisation du style de connecteur Information Engineering.



Notation

L'exemple de modèle est défini à l'aide du profil UML d' Enterprise Architect pour Modélisation des données ; la relation entre les Tableaux utilise la notation par défaut de l'ingénierie de l'information.

L'ingénierie de l'information est l'une des trois notations supporte Enterprise Architect pour aider les modélisateurs de données à identifier la cardinalité des relations. Vous pouvez modifier la notation en sélectionnant l'option de ruban 'Conception > Diagramme > Gérer > Propriétés ', en cliquant sur la page 'Connecteurs' et en sélectionnant l'option requise dans la liste déroulante 'Notation du connecteur'. Vous changeriez très probablement la notation en IDEFX1, mais la notation UML2.1 est également disponible.

SGBD par défaut

Avant de créer un Modèle de données physiques, il est conseillé de définir le SGBD par défaut pour le projet. La définition d'un SGBD par défaut garantit que tous les nouveaux éléments de base de données créés sur diagrammes se voient automatiquement attribuer le SGBD par défaut.

Si le SGBD par défaut n'est pas défini, de nouveaux Tableaux sont créés sans SGBD attribué, ce qui limite la capacité Enterprise Architect à modéliser correctement les objets physiques. Par exemple Enterprise Architect ne parvient pas à déterminer la liste correcte des types de données pour les colonnes.

Vous pouvez définir le type de SGBD par défaut en utilisant :

- ' Démarrer > Apparence > Préférences > Préférences > Source Code Engineering > Éditeurs de Code ', ou
- 'Paramètres > Données de référence > Paramètres > Types de données de base de données ou
- 'Développer > Modélisation des données > Types de données ou
- Le deuxième champ de saisie de données dans la barre d'outils de génération de code

Note : Lors de modélisation via le Database Builder, le SGBD par défaut est défini au niveau du modèle (en tant que 'DBMS' Valeur Étiquetée par rapport au Paquetage «Base de données>>) plutôt qu'au niveau du projet, permettant ainsi une plus grande flexibilité lorsque les projets impliquent plusieurs SGBD.

Transformation DDL

La transformation DDL convertit le modèle logique en un modèle de données structuré pour se conformer à l'un des SGBD pris en charge. Le type de base de données cible est déterminé par le SGBD défini comme base de données par défaut dans le modèle (voir la rubrique d'aide *Types de données de base de données* , option « Définir par défaut »). Le modèle de données peut ensuite être utilisé pour générer automatiquement des instructions DDL à exécuter dans l'un des produits de base de données pris en charge par le système.

La transformation DDL utilise et démontre support dans le langage intermédiaire d'un certain nombre de concepts spécifiques aux bases de données.

Concepts

Concept	Effet
Tableau	Mappé un à un sur les éléments de classe. Les relations « plusieurs-à-plusieurs » sont prises en charge par la transformation, créant ainsi des Join tableaux .
Colonne	Mappé un à un sur les attributs.
Primary Key	Répertorie toutes les colonnes impliquées afin qu'elles existent dans la classe et crée une méthode Primary Key pour elles.
Foreign Key	Un type spécial de connecteur, dans lequel les sections Source et Cible répertorient toutes les colonnes impliquées de sorte que : • Les colonnes existent • Une Primary Key correspondante existe dans la classe de destination, et • La transformation crée la Foreign Key appropriée

MDG Technologie pour personnaliser les mappages par défaut

Les transformations DDL qui ciblent un nouveau SGBD défini par l'utilisateur nécessitent une MDG Technologie pour mapper les types de données PIM au nouveau SGBD cible.

Pour ce faire, créez un fichier .xml MDG Technologie nommé 'UserDBMS Types.xml', en remplaçant UserDBMS par le nom du SGBD ajouté. Placez le fichier dans le dossier EA\MDGTechnologies. Le contenu du fichier MDG Technologie doit avoir cette structure :

<MDG.Technologie version="1.0">

<Documentation id="UserdataTypes" name="Userdata Types" version="1.0" notes="Mappage Type de base de données pour UserDBMS"/>

<ModulesCode>

<CodeModule langage="Données utilisateur" notes="">

<OptionsCode>

<CodeOption name="DBTypeMapping-bigint">BIGINT</CodeOption>

<CodeOption name="DBTypeMapping-blob">BLOB</CodeOption>

<CodeOption name="DBTypeMapping-boolean">TINYINT</CodeOption>

<CodeOption name="DBTypeMapping-text">CLOB</CodeOption>

...

</OptionsCode>

</CodeModule>

</CodeModules>

</MDG.Technologie>

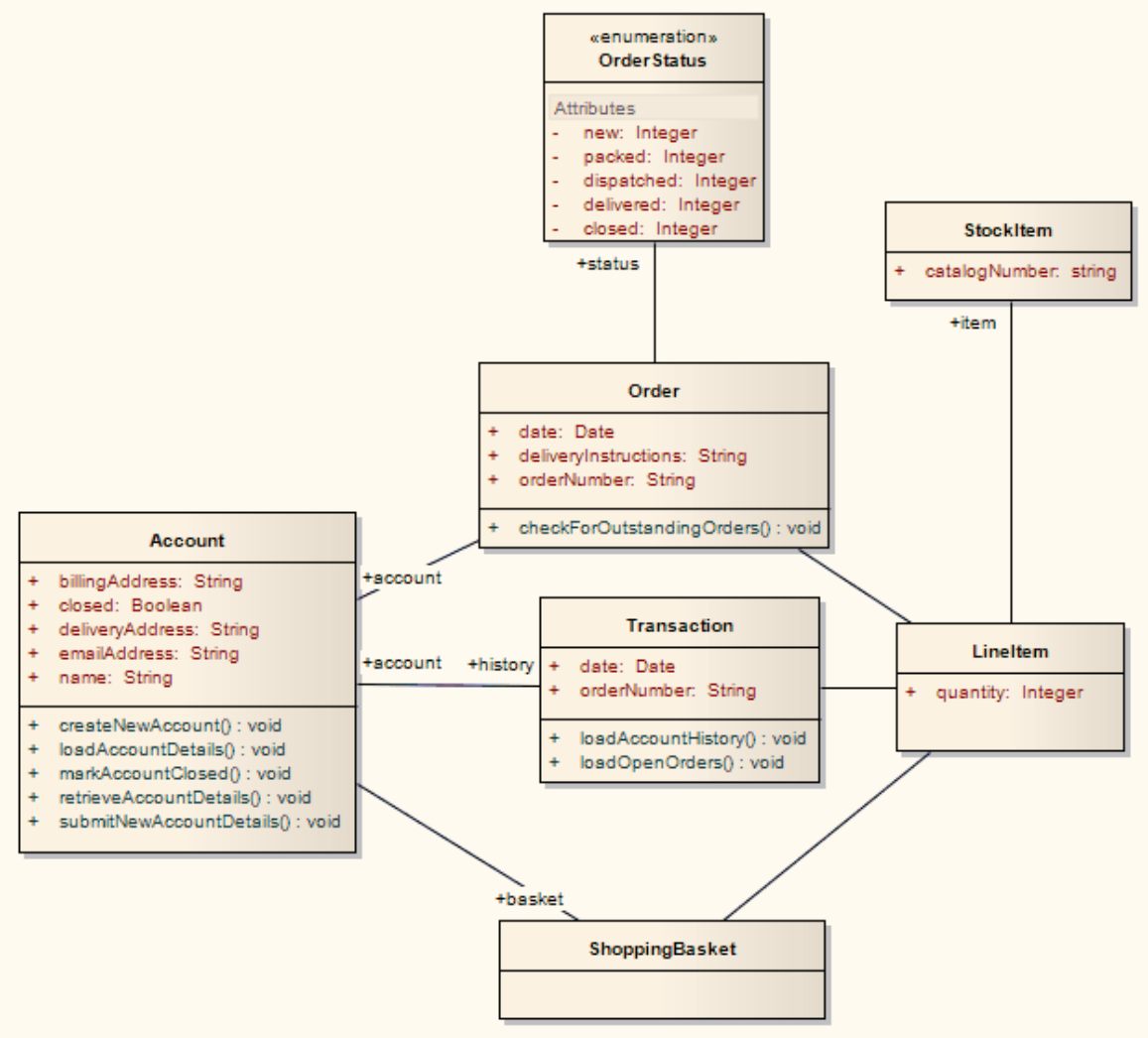
À titre d'exemple, « texte » est un Type commun (tel qu'indiqué dans la dialogue « Types de données de base de données ») qui correspond au type de données « CLOB » d'un nouveau SGBD.

Notes

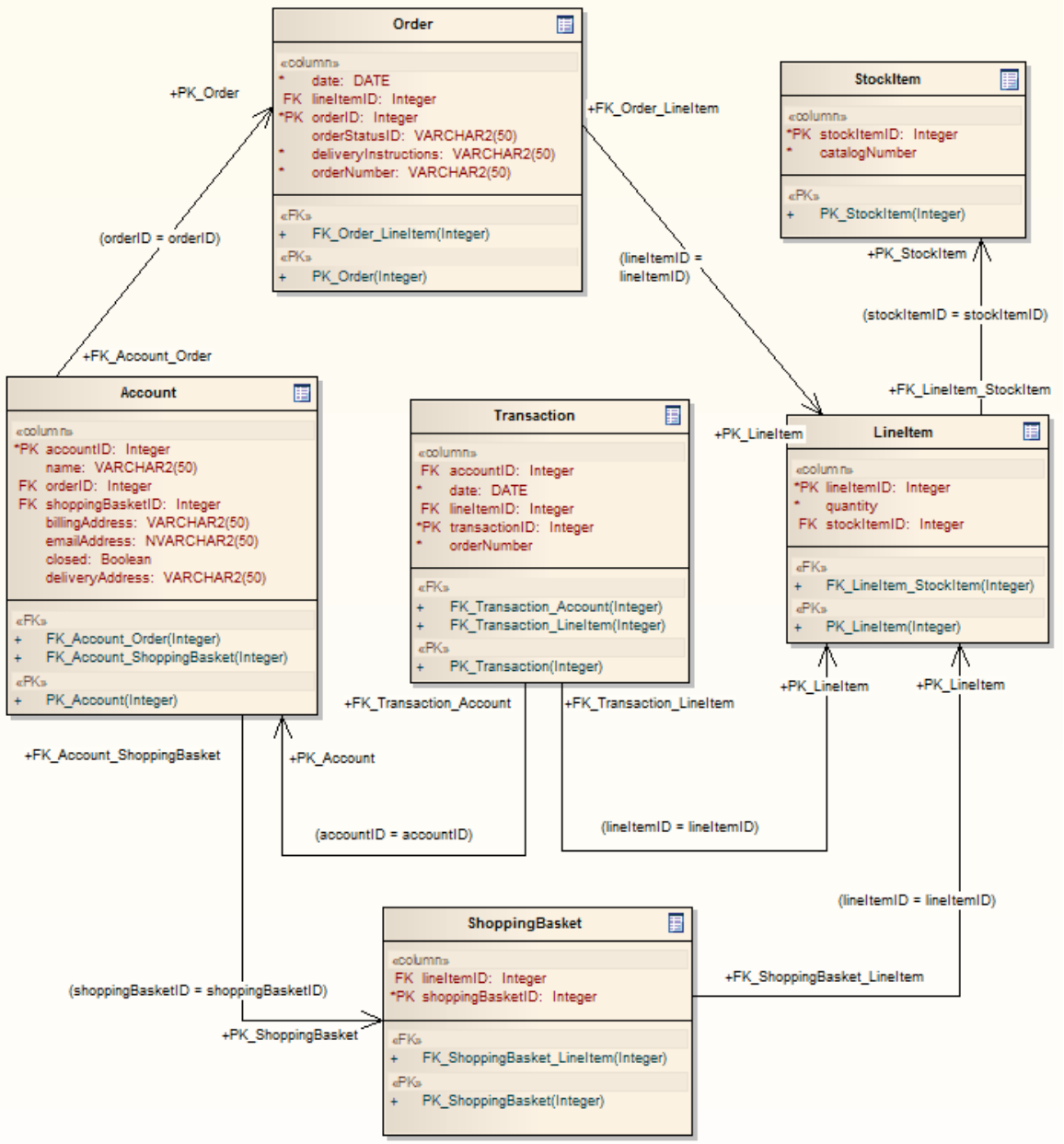
- Vous pouvez définir des aspects spécifiques au SGBD non représentés dans un modèle logique, tels que les procédures stockées, Déclencheurs , Vues et les contraintes de vérification, après la transformation ; voir la rubrique d'aide *Modèle de données physiques*

Exemple

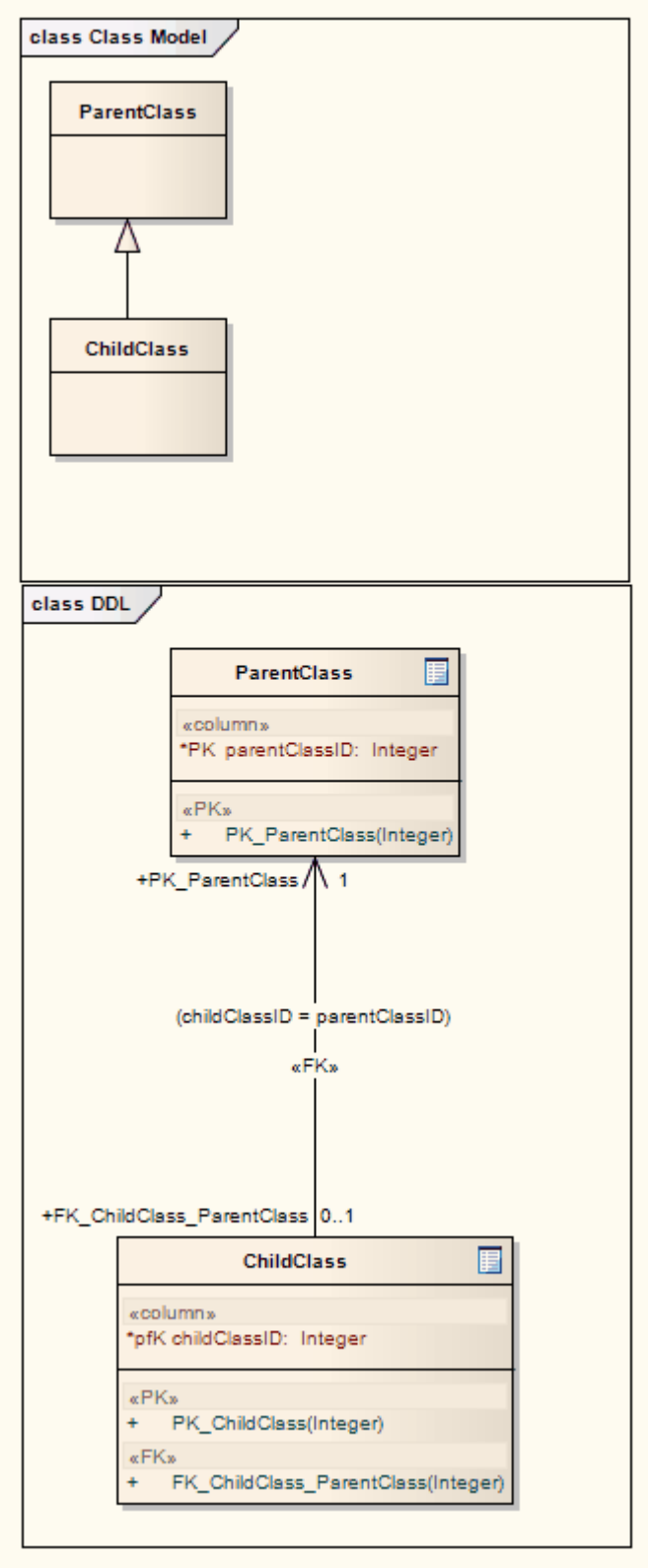
Les éléments du PIM



Après transformation, devenez les éléments PSM



Les généralisations sont gérées en fournissant à l'élément enfant une Foreign Key vers l'élément parent, comme indiqué. L'héritage par copie n'est pas pris en charge.


class Class Model
ParentClass
ChildClass
class DDL
ParentClass
«column»
*PK parentClassID: Integer
«PK»
+ PK_ParentClass(Integer)
+PK_ParentClass
1
(childClassID = parentClassID)
«FK»
+FK_ChildClass_ParentClass
0..1
ChildClass
«column»
*pfK childClassID: Integer
«PK»
+ PK_ChildClass(Integer)
«FK»
+ FK_ChildClass_ParentClass(Integer)

Plus d'information

Informations sur l'édition

Le Database Builder est disponible dans les éditions Corporate , Unified et Ultimate d' Enterprise Architect .

Systèmes de gestion de base de données pris en charge

Enterprise Architect prend en support une gamme complète de systèmes de gestion de bases de données, mais il offre également la flexibilité d'étendre le produit pour support d'autres SGBD. L'éditeur gabarit DDL peut être utilisé pour définir comment générer du DDL pour un SGBD non pris en charge, les gabarits de transformation peuvent être utilisés pour définir une nouvelle transformation en modèle physique pour un SGBD non pris en charge, et de nouveaux types de données peuvent être définis pour un SGBD existant ou nouveau. .

Enterprise Architect fournit les constructions modélisation et la possibilité de transmettre et d'effectuer une ingénierie inverse d'un schéma de base de données pour ces systèmes de gestion de base de données :

- DB2 (*)
- Oiseau de feu
- MS Access 97, 2000, 2003, 2007, 2013
- MS SQL Server à partir de 2005, toutes les éditions, y compris Express et Azure SQL Database
- MariaDB
- MySQL v4, v5
- Oracle à partir de 9i (toutes les éditions)
- PostgreSQL (y compris la version 12)
- SQLite
- Informix (#)
- Ingres (#)
- InterBase (#)
- Sybase Adaptive Server Anywhere (Sybase ASA) (#)
- Sybase Adaptive Server Entreprise (Sybase ASE) (#)

(*) - Uniquement compatible pour DB2 lorsqu'il est hébergé dans des environnements Windows et Linux.

(#) - Aucun développement supplémentaire ne sera entrepris sur ces SGBD, car ces produits ne sont pas couramment utilisés par la base d'utilisateurs Enterprise Architect . Cela permettra à Sparx Systems de concentrer ses efforts sur les autres domaines de modélisation de bases de données qui sont largement utilisés.

Notes

- Pour effectuer modélisation de données pour un SGBD particulier, vous devez disposer des types de données appropriés pour ce SGBD dans votre référentiel ; vous pouvez télécharger les définitions de données les plus récentes à partir de la page « Ressources » du site Web Sparx Systems

Générer un langage de définition de base de données (DDL)

Une fois qu'un modèle physique a été défini et les objets modélisés, Enterprise Architect peut générer un langage de définition de base de données (DDL) pour une variété d'objets, notamment Tableaux de base de données, Vues , les fonctions, les séquences et les procédures. Il s'agit d'un mécanisme permettant de gagner du temps et de réduire les erreurs qui peuvent être introduites en effectuant cette opération manuellement dans d'autres outils. L'ingénierie prospective est régie par un ensemble de gabarits qui définissent la manière dont les constructions UML sont converties en objets dans le SGBD ciblé. gabarits standard sont fournis pour tous les SGBD pris en charge, et ceux-ci peuvent être modifiés pour personnaliser la façon dont le DDL est généré. Dans le cas où un SGBD n'est pas pris en charge par défaut, un nouvel ensemble de gabarits peut être créé en utilisant ceux existants comme point de départ et de référence.

Lors de l'ingénierie avancée DDL, la sortie peut être dirigée vers un fichier (ou une série de fichiers, un pour chaque object) ou vers le moteur d'exécution DDL. Le moteur d'exécution vous permet d'exécuter le DDL immédiatement, en ciblant une base de données active via la connexion active. Si vous dirigez la sortie vers un fichier, vous pouvez exécuter le DDL sur une base de données active ultérieurement, à votre convenance. Les fichiers générés peuvent être ouverts à l'aide de l'éditeur de code, en sélectionnant F12, Ctrl+E ou Alt+7, vous permettant d'afficher le DDL dans Enterprise Architect .

Générer du DDL pour les objets

Lorsque vous créez votre modèle de base de données, vous pouvez générer le DDL pour un object individuel, un Paquetage d'objets ou le modèle de données complet. La seule différence réside dans la manière dont vous invoquez le processus de génération DDL.

Accéder

Ouvrez la fenêtre Database Builder, puis utilisez le menu contextuel et sélectionnez ' Générer DDL'.

Ruban	Développer > Modélisation de données > Database Builder > Cliquer sur un nœud object , Paquetage ou Data Modèle : Générer DDL

Onglet Générer

Champ/Bouton	Action
Paquetage	Cliquez sur le bouton et recherchez le Paquetage pour lequel vous souhaitez générer du DDL, à l'aide de la fenêtre du navigateur (une version de la dialogue « Rechercher Paquetage »). (Note : ce champ peut ne pas être affiché dans toutes les situations.)
Inclure tous Paquetages enfants	Cochez cette case pour inclure les objets des sous-Packages dans la liste 'Sélectionner les objets à Générer '.
Supprimer les fichiers cibles	Lorsque des objets sont générés dans des fichiers uniques, le nom complet du fichier est stocké avec l' object et affiché dans la colonne 'Fichier cible' de la liste 'Sélectionner les objets à Générer '. Cliquez sur ce bouton pour supprimer tous les noms de fichiers existants et en prompt de nouveaux.
Sélectionner les objets à Générer	Ce champ affiche la liste des objets pour lesquels le DDL sera généré, dans l'ordre affiché. Si vous devez modifier cet ordre pour résoudre les dépendances object , cliquez sur un object à déplacer et cliquez sur les boutons pour déplacer cet object d'une position vers le haut ou vers le bas dans la séquence. Sélectionnez chaque object pour lequel générer du DDL. Cliquer sur: • Le bouton Tous pour sélectionner chaque élément • Le bouton Aucun pour effacer toutes les sélections • Chacun de plusieurs objets pendant que vous appuyez sur Ctrl, pour sélectionner un certain nombre d'objets individuels • Le premier et le dernier objets d'un bloc pendant que vous appuyez sur Maj, pour sélectionner chaque object du bloc
Enregistrer la commande générée	Si vous avez modifié l'ordre dans lequel les objets sont listés, cochez la case pour enregistrer la nouvelle séquence lorsque vous cliquez sur le bouton Générer .

Rafraîchir	Rechargez la liste des objets, en restaurant chaque object à sa position précédente (si la position object a été modifiée).
Un seul fichier	Sélectionnez ce bouton radio si vous souhaitez enregistrer le DDL généré dans un seul fichier. Cliquez sur le bouton pour rechercher le chemin et le nom du fichier.
Fichier individuel pour chaque tableau	Sélectionnez ce bouton radio si vous souhaitez enregistrer le DDL généré pour chaque object dans un fichier distinct. Lorsque vous cliquez sur le bouton Générer , le système vous prompts tour à tour le nom du fichier cible pour chaque object (s'il n'est pas déjà spécifié).
Générer vers DDL Moteur d'Exécution	Sélectionnez ce bouton radio si vous souhaitez enregistrer le DDL dans le moteur d'exécution (l'onglet « Exécuter le DDL » du générateur de base de données). Le Moteur d'Exécution DDL fournit les facilités pour exécuter le script SQL généré et répondre immédiatement aux erreurs d'exécution, sans avoir à créer un fichier externe et à le charger dans un autre outil. ' Générer vers DDL Moteur d'Exécution ' est l'option par défaut si le Database Builder est ouvert.
Générer	Cliquez sur ce bouton pour exécuter le processus Générer DDL avec les options que vous avez sélectionnées.
Vue	Si vous avez généré le DDL dans un seul fichier externe, cliquez sur ce bouton pour afficher la sortie. Par défaut Enterprise Architect utilise l'éditeur de code par défaut. Cependant, vous pouvez définir un autre éditeur DDL par défaut dans la dialogue 'Préférences' (' Démarrer > Application > Préférences > Préférences > Source Code Engineering > Éditeurs de Code > DDL').
Fermer	Cliquez sur ce bouton pour fermer le dialogue . Si vous n'avez pas généré le DDL, ce bouton abandonne également la génération du DDL pour l' object .

Onglet Options

Définissez l'un de ces indicateurs sur False si vous ne souhaitez pas entreprendre l'action qu'ils lancent.

Groupe	Possibilités
Options de génération Tableau	**Tableaux** - indique que le DDL pour les éléments Tableau doit être généré (*) **Primary Keys** - indique que le DDL pour Primary Keys doit être généré ($) **Foreign Keys** - indique que le DDL pour Foreign Keys doit être généré ($) **Index** - indique que le DDL pour les index doit être généré ($) **Contraintes uniques** - indique que le DDL pour les contraintes uniques doit être généré ($) **Vérifier les contraintes** - indique que le DDL pour les contraintes de vérification doit être généré ($) **Tableau Déclencheurs** - indique que le DDL pour Tableau Déclencheurs doit être généré ($)

	Propriétés Tableau : indique que le DDL pour les propriétés tableau étendues doit être généré ($) **Sémantique de longueur** - indique que le DDL pour Oracle Longueur Sémantique doit être généré ($)
Options de génération Object	**Vues** - indique que le DDL pour les éléments Vue doit être généré (*) **Procédures** - indique que le DDL pour les éléments de procédure doit être généré (*) **Fonctions** - indique que le DDL pour les éléments de fonction doit être généré (*) **Séquences** - indique que le DDL pour les éléments Séquence doit être généré (*) **Paquetages** - indique que le DDL pour les éléments Oracle Paquetages doit être généré (*)
Mise en page	**Inclure les requêtes pré/post** - indique que le DDL généré doit inclure les instructions SQL définies dans les requêtes SQL '_PreStatements' et '_PostStatements' **Inclure les propriétaires** - indique que le DDL généré doit inclure le schéma/propriétaire de tous les éléments **Inclure les commentaires** : indique que le DDL généré doit inclure des commentaires. **Inclure les commentaires d'en-tête** - indique que le DDL généré doit inclure tous les commentaires d'en-tête (#) **Inclure les commentaires Object** - indique que le DDL généré doit inclure tout commentaire object (tel que Tableau ou Vue) (#) **Inclure les commentaires de colonne** - indique que le DDL généré doit inclure tous les commentaires de colonnes (#) **Générer des instructions DROP** - indique que le DDL généré doit inclure l'instruction DROP pour les objets **Utiliser la base de données** : indique que le DDL généré doit inclure une instruction USE Database **Utiliser un alias** : indique que le DDL généré utilise n'importe quel alias object ou de colonne. **Contrainte séparée de Tableau** : indique que le DDL généré doit définir la création de contraintes en tant qu'instructions distinctes de la définition Tableau . **Inclure NULL dans les définitions de colonne** : indique que le DDL généré doit appliquer le mot clé NULL à chaque définition de colonne définie comme nullable ; c'est-à-dire les colonnes avec leur indicateur 'NOT NULL' décoché (cette option s'applique uniquement aux SGBD qui support la syntaxe 'NULL')

Notes

- (*) - les options avec cette marque seront automatiquement définies sur True si vous avez spécifié de générer du DDL pour un élément individuel de ce type ; c'est-à-dire que si vous sélectionnez un Tableau et que votre option ' Générer Tableau ' est False, Enterprise Architect changera l'option en True
- ($) - les options avec cette marque seront désactivées si l'option ' Tableaux ' est définie sur False
- (#) - les options portant cette marque seront désactivées si l'option « Inclure les commentaires » est définie sur False.
- Dans les éditions Corporate , Unified et Ultimate d' Enterprise Architect , si la sécurité est activée, vous devez disposer de l'autorisation « Générer Code Source et DDL » pour générer du DDL.
- Pour une base de données PostgreSQL, vous devez définir l'option « Séquences » sur True pour permettre la création

de colonnes à incrémentation automatique.

- Si vous générez des séquences Oracle, vous devez toujours définir les options ' Tableau Déclencheurs ' et ' Séquences ' sur True, afin qu'un déclencheur de pré-insertion soit généré pour sélectionner la valeur de séquence suivante pour remplir la colonne ; également, dans les propriétés de la colonne, définissez la propriété 'AutoNum' sur True
- Vous pouvez modifier les gabarits DDL que le système utilise pour générer le DDL ; ceux-ci sont stockés au niveau du référentiel afin que tous les autres utilisateurs du même référentiel utilisent automatiquement les gabarits mis à jour

Modifier Gabarits DDL

L'éditeur DDL Gabarit offre la possibilité de modifier les gabarits que le système utilise lors de la génération de DDL à partir d'un modèle de données. Il applique les facilités du Common Éditeur de Code , y compris Intelli-sense pour les différentes macros. Pour plus d'informations sur Intelli-sense et Common Éditeur de Code , consultez la rubrique d'aide *Modification du code source* .

Accéder

Ruban	Développer > Modélisation des données > Gabarits

Sélectionner et modifier Gabarits

Option	Action
Langue	Cliquez sur la flèche déroulante et sélectionnez le type de base de données (Database Management System).
Nouvelle base de données	Cliquez sur ce bouton pour créer un nouvel ensemble de gabarits pour un SGBD non standard. La dialogue 'Saisie' s'affiche, sur laquelle vous tapez le nom du nouveau SGBD pour lequel vous créez gabarits . Cela met à jour le champ « Langue ».
Gabarit	Affiche le contenu du gabarit sélectionné et fournit l'éditeur pour modifier ce contenu.
Gabarits	Répertorie les gabarits DDL de base. Cliquez sur le nom d'un gabarit pour afficher et modifier le contenu gabarit ; le nom du gabarit sélectionné est mis en surbrillance. Le champ 'Modifié' indique si vous avez modifié le gabarit par défaut initialement fourni avec le système.
Remplacement des stéréotypes	Répertorie tous gabarits stéréotypés qui existent pour le gabarit de base actuellement sélectionné. Le champ 'Modifié' indique si vous avez modifié un gabarit stéréotypé par défaut.
Ajouter un nouveau Gabarit personnalisé	Cliquez sur ce bouton pour afficher la dialogue 'Créer un nouveau Gabarit personnalisé', dans laquelle vous sélectionnez le type gabarit dans une liste déroulante et saisissez un nom pour le gabarit . Le type gabarit devient un préfixe pour le nom ; Par exemple: Espace de noms_MyDDLTemplate
Ajouter un nouveau remplacement stéréotypé	Sélectionnez un gabarit de base et cliquez sur ce bouton pour afficher la dialogue 'New Gabarit Override' permettant d'ajouter un gabarit stéréotypé pour le gabarit sélectionné. Dans les listes déroulantes, sélectionnez le stéréotype Classe et/ou

	Fonctionnalité pour lequel appliquer le gabarit de remplacement.
Obtenir Gabarit par défaut	Cliquez sur ce bouton pour rafraîchir l'affichage de l'éditeur avec la version par défaut du gabarit sélectionné. (Cela ne supprime pas la version modifiée du gabarit .)
Sauvegarder	Cliquez sur ce bouton pour écraser le gabarit sélectionné avec le contenu mis à jour du panneau Gabarit .
Supprimer	Si vous avez remplacé le gabarit sélectionné, cliquez sur ce bouton pour supprimer le gabarit remplacé et le remplacer par le gabarit DDL par défaut correspondant.

Notes

- Gabarits DDL modifiés et définis par l'utilisateur peuvent être importés et exportés en tant que données de référence (voir la rubrique *Partage des données de référence*)
- Tous gabarits définis par l'utilisateur pour un type de base de données sont répertoriés dans la dialogue 'Exporter les données de référence' du tableau 'Code, DDL, Transformation & CSV Gabarits ', identifiés par le nom du SGBD avec le suffixe _DDL_Template - si aucun gabarits défini par l'utilisateur n'existe. pour un SGBD, il n'y a pas d'entrée pour le SGBD dans le dialogue
- Vous devez également définir les types de données appropriés pour le SGBD et, si vous exportez les gabarits en tant que données de référence, vous devez également exporter le tableau 'Types de données Modèle - Code et DDL'.

Syntaxe DDL Gabarit

Gabarits DDL sont écrits à l'aide du framework Code Gabarit d' Enterprise Architect , mais ils ont été étendus pour support la génération DDL.

Développement DDL Gabarit

Ces aspects du développement de DDL Gabarit sont abordés dans cette section.

Aspect	Voir également
Gabarits DDL	Gabarits DDL
Macros DDL	Macros DDL
Macros de fonctions DDL	Macros de fonctions DDL
Macros de propriétés DDL	Macros de propriétés DDL
Options DDL dans Gabarits	Options DDL dans Gabarits

Gabarits DDL

L'éditeur DDL Gabarit fonctionne de la même manière que l'éditeur Code Gabarit , sauf que l'éditeur DDL Gabarit affiche simultanément gabarits pour la génération DDL et gabarits pour la génération Alter DDL. Les gabarits Alter DDL Generation sont affichés en bas de la liste, préfixés par 'DDL Diff'.

Gabarits de base pour la génération DDL

Le framework DDL Gabarit se compose d'un certain nombre de gabarits de base pour la génération DDL. Chaque gabarit de base génère une instruction DDL (ou une instruction partielle) pour un aspect particulier du modèle de données UML .

Gabarits

Ce tableau répertorie et décrit brièvement les gabarits de base utilisés pour la génération DDL.

Gabarit	Description
Contrainte de vérification DDL	Invoqué par le gabarit de contrainte Tableau DDL pour générer les instructions de contrainte de vérification pour un object Tableau .
Commentaire de colonne DDL	Normalement invoqué par le DDL Créez gabarit Tableau Extras pour générer des instructions COMMENT ON (ou équivalent) pour chaque colonne Tableau .
Définition de la colonne DDL	Invoqué par de nombreux gabarits pour construire l'instruction permettant de créer une seule colonne Tableau , telle qu'elle apparaît dans l'instruction CREATE TABLE.
Extras de colonne DDL	Normalement invoqué par le gabarit DDL Create Tableau Extras pour générer les propriétés de colonne étendues pour chaque colonne Tableau .
Nom de colonne de contrainte DDL	Invoqué par chacun des gabarits de contrainte pour récupérer les noms de colonnes correctement formatés impliqués dans la contrainte actuelle.
Commentaire sur la contrainte DDL	Normalement invoqué par le gabarit DDL Create Tableau Extras pour générer des instructions COMMENT ON (ou équivalentes) pour chaque contrainte Tableau .
DDL Créer Foreign Keys	Invoqué par le DDL Créer des contraintes Tableau gabarit pour générer des contraintes Foreign Key pour un object Tableau .
Fonction de création DDL	Invoqué par le fichier de script DDL gabarit pour générer l'instruction CREATE FUNCTION pour un object Function.
DDL Créer Paquetage	Invoqué par le fichier de script DDL gabarit pour générer les instructions CREATE PACKAGE pour un object Paquetage (Oracle uniquement).
Procédure de création DDL	Invoqué par le fichier de script DDL gabarit pour générer l'instruction CREATE PROCEDURE pour un object Procedure .
DDL Créer un schéma	Actuellement non utilisé.
Séquence de création DDL	Invoqué par le fichier de script DDL gabarit pour générer l'instruction CREATE SEQUENCE pour un object Séquence .
DDL Créer Tableau	Invoqué par le fichier de script DDL gabarit pour générer l'instruction CREATE TABLE pour un object Tableau .
DDL Créer des contraintes	Invoqué par le fichier de script DDL gabarit pour générer des contraintes Tableau et

Tableau	des index pour un object Tableau .
DDL Créer des extras Tableau	Invoqué par le fichier de script DDL gabarit pour générer des propriétés Tableau étendues pour un object Tableau .
DDL Créer Vue	Invoqué par le fichier de script DDL gabarit pour générer l'instruction CREATE VIEW pour un object Vue .
Type de données DDL	Invoqué par le gabarit de définition de colonne DDL pour générer l'instruction de type de données correctement formatée pour une colonne Tableau .
Extras de colonne de suppression DDL	Invoqué par le gabarit DDL Drop Tableau Extras pour générer des instructions drop spécialisées pour les propriétés étendues de colonne.
DDL supprime Foreign Keys	Invoqué par le fichier de script DDL gabarit pour générer les instructions pour DROP toutes Foreign Keys pour un object Tableau .
Fonction de suppression DDL	Invoqué par le fichier de script DDL gabarit pour générer l'instruction DROP FUNCTION pour un object Function.
Procédure de suppression DDL	Invoqué par le fichier de script DDL gabarit pour générer l'instruction DROP PROCEDURE pour un object Procedure .
Séquence de dépôt DDL	Invoqué par le fichier de script DDL gabarit pour générer l'instruction DROP SEQUENCE pour un object Séquence .
DDL Supprimer Tableau	Invoqué par le fichier de script DDL gabarit pour générer l'instruction DROP TABLE pour un object Tableau .
DDL supprime les extras Tableau	Invoqué par le gabarit du fichier de script DDL pour générer les instructions permettant de SUPPRIMER toutes les propriétés étendues d'un object Tableau .
DDL Supprimer Vue	Invoqué par le gabarit du fichier de script DDL pour générer l'instruction DROP VIEW pour un object Vue .
Contrainte étrangère DDL	Invoqué par le gabarit de contraintes Tableau DDL pour générer les instructions ADD FOREIGN KEY CONSTRAINT pour un object Tableau .
Subvention DDL	Invoqué par le gabarit DDL Create Tableau Extras pour générer l'instruction GRANT pour l' object actuel (Oracle uniquement).
Index DDL	Invoqué par le gabarit de contrainte DDL Tableau pour générer les instructions CREATE INDEX pour un object Tableau .
Surround gauche DDL	Utilisé pour définir le ou les caractères utilisés comme délimiteur gauche sur le nom d'un object (ou d'un composant object).
Nom DDL	Utilisé par la plupart gabarits pour fournir une manière courante de formater le nom d'un object (ou fonctionnalité object). Ce gabarit accepte quatre paramètres : • Emplacement Object (valeurs : EA ou LIVE) • Type Object (valeurs : OWNER, TABLE, VIEW, PROCEDURE, FUNCTION, SEQUENCE, PACKAGE, COLUMN, CONSTRAINT, CONSTRAINT_COLUMN, REFERENCE_TABLE, REFERENCE_COLUMN)

	• Inclure le drapeau du propriétaire ; contrôle si le nom doit être préfixé par le nom du propriétaire (valeurs : INCLUDE_OWNER ou {blank}) • Inclure le drapeau Surround ; contrôle si le nom doit être délimité par les caractères surround gauche et droit (valeurs : INCLUDE_SURROUND ou {blank})
Contrainte principale DDL	Invoqué par le gabarit de contrainte DDL Tableau pour générer l'instruction ADD PRIMARY KEY CONSTRAINT pour un object Tableau .
Nom de la colonne de référence DDL	Normalement invoqué par les gabarits de noms DDL pour récupérer les noms de colonnes de référence correctement formatés impliqués dans une Foreign Key .
Définition de référence DDL	Invoqué par le gabarit de contrainte étrangère DDL pour générer les instructions ON DELETE/ON UPDATE pour une contrainte Foreign Key .
Surround droit DDL	Permet de définir le ou les caractères utilisés comme délimiteur droit sur le nom d'un object (ou d'un composant object).
Fichier de script DDL	Un gabarit de haut niveau pour générer du DDL ; tous les autres gabarits sont invoqués à partir de celui-ci.
En-tête de script DDL	Invoqué par le fichier de script DDL gabarit pour ajouter un commentaire d'en-tête au début de chaque fichier DDL.
Séparateur de script DDL	Utilisé par tous gabarits qui doivent inclure un séparateur d'instructions dans le DDL généré.
Durée de la déclaration DDL	Utilisé pour définir le ou les caractères utilisés comme terminateur d'instruction. Par exemple, un point-virgule (';') pour la plupart des SGBD.
Terme de l'instruction DDL Alt	Utilisé pour définir le ou les caractères utilisés comme terminateur d'instruction alternatif. Par exemple, dans certains SGBD, le terminateur d'instruction doit être modifié afin de ne pas causer de problèmes avec les instructions DDL générées pour les objets basés sur SQL, tels que Vues et les procédures.
DDL Synonyme	Appelé par le gabarit DDL Create Tableau Extras pour générer l'instruction CREATE SYNONYMS (Oracle uniquement).
Contrainte Tableau DDL	Invoqué par le gabarit DDL Create Tableau Constraints pour générer les contraintes et index Tableau pour chaque object Tableau , en tenant compte des options de génération.
Commentaire au niveau Tableau DDL	Invoqué par le gabarit DDL Create Tableau Extras pour générer des instructions COMMENT ON (ou l'équivalent) pour un object .
DDL Déclencheur	Invoqué par la contrainte DDL Tableau gabarit pour générer les instructions CREATE TRIGGER pour un object Tableau .
Contrainte unique DDL	Invoqué par le gabarit de contrainte DDL Tableau pour générer les instructions ADD UNIQUE CONSTRAINT pour un object Tableau .
Base de données d'utilisation DDL	Invoqué par le gabarit du fichier de script DDL pour inclure une instruction USE DATABASE au début de chaque fichier DDL.

Gabarits de base pour la génération Alter DDL

Le framework DDL Gabarit se compose d'un certain nombre de gabarits de base pour la génération Alter DDL. Chaque gabarit de base génère une ou plusieurs instructions DDL basées sur l' *Action* détectée qui doit être entreprise pour synchroniser le modèle de données et la base de données en direct.

Gabarits

Ce tableau répertorie et décrit brièvement les gabarits de base utilisés pour la génération Alter DDL.

Gabarit	Description
Colonne DDL Diff	Invoqué directement par Enterprise Architect pour chaque différence de colonne Tableau détectée.
Contrainte Diff DDL	Invoqué directement par Enterprise Architect pour chaque différence de contrainte Tableau détectée.
Tableau des différences DDL	Invoqué directement par Enterprise Architect pour chaque différence Tableau détectée.
Vue DDL Diff	Invoqué directement par Enterprise Architect pour chaque différence Vue détectée.
Procédure de différence DDL	Invoqué directement par Enterprise Architect pour chaque différence de procédure stockée détectée.
Fonction DDL Diff	Invoqué directement par Enterprise Architect pour chaque différence de fonction détectée.
Séquence Diff DDL	Invoqué directement par Enterprise Architect pour chaque différence Séquence détectée.

Macros DDL

Les macros de substitution de champs permettent d'accéder aux données de votre modèle. Ils sont notamment utilisés pour accéder aux champs de données depuis :

- Objets de base de données (tels que Tableaux et Vues)
- Colonnes
- Contraintes
- Colonnes de contrainte

Les macros de substitution de champ sont nommées selon la casse Camel. Par convention, toutes les macros DDL portent le préfixe « ddl ».

Les macros qui représentent des cases à cocher ou des valeurs booléennes renvoient une valeur string de « T » si la case à cocher/booléen est vraie. Sinon, une string vide est renvoyée.

Macro de champ interne - ddlAction

La macro ddlAction est une macro interne disponible dans les gabarits 'Alter DDL', offrant un accès direct aux champs internes d' Enterprise Architect ; il n'a aucun mappage direct avec les données stockées.

ddlAction représente l'action qui doit être entreprise pour synchroniser la base de données active avec le référentiel actuel. Par exemple, « Créer Tableau », « Supprimer Tableau » ou « Changer de propriétaire ».

Macros de champs d'éléments

Cette liste identifie les macros disponibles dans gabarits DDL pour accéder aux champs au niveau des éléments, où (dans Enterprise Architect) les champs sont modifiables, tels que « Nom Tableau » et « Alias Tableau ».

ddlFunctionAlias

Boîte dialogue ' Propriétés ' de la fonction : Onglet 'Principal' : 'Alias' champ de texte.

ddlFunctionName

dialogue ' Propriétés ' de la fonction : champ de texte 'Nom'.

ddlOwner

{ Élément Tableau } dialogue ' Propriétés ' : {élément} Onglet ' Détails Tableau ' : champ de texte ' Propriétaire '.

ddlPackageAlias

Boîte dialogue ' Propriétés ' Paquetage : onglet 'Principal' : 'Alias' champ de texte.

ddlPackageName

Boîte dialogue ' Propriétés ' Paquetage : champ de texte 'Nom'.

ddlProcedureAlias

Procédure Boîte dialogue ' Propriétés ' : Onglet 'Principal' : 'Alias' champ de texte.

ddlProcedureName

Procédure dialogue ' Propriétés ' : champ de texte 'Nom'.

ddlSchemaFunctionName

Le nom de la définition de l’élément Function lu dans la base de données active.

ddlSchemaOwner

La propriété « Propriétaire » de la définition de l'élément lue à partir de la base de données active.

ddlSchemaProcedureName

Le nom de la définition de l'élément Procedure lu dans la base de données active.

ddlSchemaSequenceName

Le nom de la définition de l'élément Séquence lu dans la base de données en direct.

ddlSchemaTableName

La propriété « Tableau Name » est lue à partir de la base de données active.

ddlSchemaViewName

Le nom de la définition de l'élément Vue lu dans la base de données en direct.

ddlSequenceAlias

Séquence dialogue ' Propriétés ' : Onglet 'Principal' : champ de texte 'Alias'.

ddlSequenceName

Séquence dialogue ' Propriétés ' : champ de texte 'Nom'.

ddlTableAlias

Boîte dialogue Tableau ' Propriétés ' : onglet 'Principal' : 'Alias' champ de texte.

ddlTableDBMS

dialogue Tableau ' Propriétés ' : onglet 'Principal' : champ de liste déroulante 'Base de données'.

ddlTableLevelComment

dialogue Tableau ' Propriétés ' : champ de texte ' Notes '.

ddlTableName

dialogue Tableau ' Propriétés ' : champ de texte 'Nom'.

ddlViewAlias

Boîte dialogue ' Propriétés ' Vue : Onglet 'Principal' : 'Alias' champ de texte.

ddlViewName

dialogue ' Propriétés ' Vue : champ de texte 'Nom'.

Macros de champs de colonnes

Cette liste identifie les macros disponibles dans gabarits DDL pour accéder aux champs liés aux colonnes, où (dans Enterprise Architect) les champs sont modifiables, tels que « Nom de la colonne » et « Alias de la colonne ».

ddlColumnName

Boîte dialogue 'Colonnes et contraintes' : Onglet 'Colonne' : 'Nom' cellule.

ddlColumnAlias

dialogue 'Colonnes et Contraintes' : Onglet 'Colonne' : cellule 'Alias'.

ddlColumnComment

dialogue 'Colonnes et Contraintes' : Onglet 'Colonne' : champ de texte ' Notes '.

ddlSchemaColumnName

Propriété Column **Name** lue à partir de la base de données active.

Note : Ce champ n'est pas modifiable directement dans Enterprise Architect .

Macros de champs de contraintes

Ce tableau répertorie les macros disponibles dans gabarits DDL pour accéder aux champs liés aux contraintes, où (dans Enterprise Architect) les champs sont modifiables, tels que 'Nom de la contrainte' et ' Type de contrainte'.

ddlConstraintAlias

Boîte dialogue 'Colonnes et Contraintes' : Onglet 'Contraintes' : cellule 'Alias'.

ddlConstraintColumnAlias

dialogue 'Colonnes et contraintes' : onglet 'Contraintes' : liste 'Colonnes impliquées : attribuées'.

ddlConstraintColumnName

dialogue 'Colonnes et contraintes' : onglet 'Contraintes' : liste 'Colonnes impliquées : attribuées'.

ddlConstraintComment

Boîte dialogue 'Colonnes et contraintes' : onglet 'Contraintes' : champ de texte ' Notes '.

ddlConstraintName

dialogue 'Colonnes et contraintes' : Onglet 'Contraintes' : 'Nom' cellule.

ddlPKColumnCount

Uniquement pertinent si la contrainte actuelle a un type de Primary Key , cette macro renverra un nombre de colonnes attribuées à la Primary Key .

dialogue 'Colonnes et contraintes' : onglet 'Contraintes' : liste 'Colonnes impliquées : attribuées'.

ddlReferenceColumnAlias

Uniquement pertinent si la contrainte actuelle a un type de Foreign Key , cette macro renverra l'alias de colonne du tableau de référence.

Boîte dialogue 'Colonnes et contraintes' : Onglet 'Contraintes' : 'Alias' cellule.

ddlReferenceColumnName

Uniquement pertinent si la contrainte actuelle a un type de Foreign Key , cette macro renverra le nom de la colonne du tableau de référence.

dialogue 'Contrainte' Foreign Key : liste 'Colonnes impliquées' : colonne 'Parent'.

ddlReferenceTableAlias

Uniquement pertinent si la contrainte actuelle a un type de Foreign Key , cette macro renverra l'alias du tableau de référence.

Boîte dialogue Tableau ' Propriétés ' : onglet 'Principal' : 'Alias' champ de texte.

ddlReferenceTableName

Uniquement pertinent si la contrainte actuelle a un type de Foreign Key , cette macro renverra le nom du tableau de référence.

dialogue 'Contrainte' Foreign Key : liste 'Colonnes impliquées' : en-tête de colonne 'Parent'.

ddlReferenceTableOwner

Uniquement pertinent si la contrainte actuelle a un type de Foreign Key , cette macro renverra le propriétaire du tableau de référence.

dialogue 'Contrainte' Foreign Key : liste 'Colonnes impliquées' : en-tête de colonne 'Parent'.

ddlSchemaConstraintColumnName

Les noms de colonnes impliqués dans la contrainte actuelle sont lus à partir de la base de données active.

Note : ce champ n'est pas modifiable directement dans Enterprise Architect .

ddlSchemaConstraintName

La propriété Constraint **Name** est lue à partir de la base de données active.

Note : ce champ n'est pas modifiable directement dans Enterprise Architect .

ddlSchemaConstraintType

La propriété **Type** de contrainte est lue à partir de la base de données active.

Note : ce champ n'est pas modifiable directement dans Enterprise Architect .

Macros de fonctions DDL

Les macros de fonction DDL offrent un moyen pratique de manipuler, récupérer ou formater les données d'éléments pertinentes pour la génération DDL. Ces macros, ainsi que les macros de fonctions de code, sont disponibles pour les gabarits DDL. Chaque macro de fonction renvoie une string de résultat et est utilisée de la même manière qu'une macro de fonction Code Gabarit .

Les macros de fonctions disponibles sont décrites ici. Tous les paramètres ont un type de String et sont indiqués par des crochets ; c'est-à-dire : FUNCTION_NAME([param]).

DDL_DATATYPE_SIZE ([productName] , [datatype])

Renvoie le type de données entièrement formaté de la colonne actuelle dans la syntaxe DDL.

Paramètres

- productName - le SGBD attribué à Tableau actuel, tel que SQL Server 2012, Oracle ou PostgreSQL
- datatype - le nom du type de données de la colonne actuelle, tel que VARCHAR ou INT

Remarques

Dans une colonne Tableau Enterprise Architect , les types de données sont définis avec une propriété Type de longueur (0, 1 ou 2) qui influence la syntaxe DDL ; cette macro de fonction prend en compte le Type de longueur (et d'autres facteurs) lors de la création de la valeur de retour.

DDL_GET_DEFINITION_PARAS ([definition])

Renvoie une représentation string des paramètres de la définition de fonction/procédure fournie.

Paramètres

- définition - la définition SQL complète de la procédure/fonction

Remarques

Certains SGBD (tels que PostgreSQL) support plusieurs définitions du même nom de procédure/fonction. Les définitions diffèrent uniquement par leur liste de paramètres, donc pour manipuler de tels objets, le DDL doit spécifier le nom et les paramètres. Cette macro de fonction donne aux gabarits DDL la possibilité d'extraire les paramètres afin qu'ils puissent ensuite être utilisés pour identifier des objets individuels.

DDL_INCLUDE_SQLQUERY ([objectName])

Renvoie l'instruction SQL définie dans l' object SQLQuery.

Paramètres

- objectName - le nom de l' object Query SQL défini dans le modèle de données actuel

Remarques

Aucun.

DDL_INDEX_SORT ([product] , [columns])

Renvoie l'ordre de tri d'un index donné.

Paramètres

- produit - le SGBD (actuellement, Firebird)
- columns - un CSV des noms de colonnes impliqués dans l'index

Remarques

Cette macro ne s'applique actuellement qu'aux index Firebird.

DDL_RESOLVE_NAME ([productName] , [name] , [leftSurround] , [rightSurround])

Renvoie le nom fourni délimité (avec les caractères gauche et droit fournis) si le nom est un mot réservé pour le SGBD actuel.

Paramètres

- productName - le SGBD attribué à Tableau actuel, tel que SQL Server 2012, Oracle ou PostgreSQL
- name - le nom de l' object /de la colonne
- leftSurround - le caractère gauche de la paire utilisée pour entourer le nom ; par exemple, guillemet simple {'}
- rightSurround - le caractère droit de la paire utilisée pour entourer le nom ; par exemple, guillemet simple {'}

Remarques

La syntaxe DDL de certains SGBD nécessite que les noms qui sont des mots réservés soient délimités d'une manière différente ; cette macro de fonction peut être utilisée pour formater en toute sécurité tous les noms pour DB2 et Firebird.

DDL_TABLE_TAGVALUE ([tagName])

Renvoie la valeur du nom de balise fourni dans la version du référentiel du Tableau actuel.

Paramètres

- tagName - le nom de l'élément de balise à récupérer

Remarques

Aucun.

EXECUTE_CURRENT ([objectName] , [actionName] , [priority])

Ajoute la string de retour du gabarit courant à la file d'attente d'exécution du Moteur d'Exécution .

Paramètres

- objectName - la valeur qui sera affichée dans la colonne ' Object ' de la file d'attente d'exécution, qui indique le nom de l' object en cours de mise à jour
- actionName - la valeur qui sera affichée dans la colonne ' Action ' de la file d'attente d'exécution, qui indique l'action qui a abouti à la génération de cette instruction
- priorité - une valeur numérique qui représente la priorité de l'instruction ; plus le nombre est élevé, plus l'instruction est placée bas dans la file d'attente

Remarques

Cette macro de fonction peut être appelée à tout moment dans le gabarit , mais ne s'exécutera qu'à la fin. Une fois le gabarit terminé, le DDL qu'il a généré est envoyé dans la file d'attente d'exécution.

Cette macro de fonction n'a aucun effet si l'utilisateur a choisi de générer du DDL dans un fichier.

EXECUTE_STRING ([objectName] , [actionName] , [priority] , [ddlStatement])

Ajoute l'instruction DDL fournie à la file d'attente d'exécution du Moteur d'Exécution .

Paramètres

- objectName - la valeur qui sera affichée dans la colonne ' Object ' de la file d'attente d'exécution, qui indique le nom de l' object en cours de mise à jour
- actionName - la valeur qui sera affichée dans la colonne ' Action ' de la file d'attente d'exécution, qui indique l'action qui a abouti à la génération de cette instruction
- priorité - une valeur numérique qui représente la priorité de l'instruction ; plus le nombre est élevé, plus l'instruction est placée bas dans la file d'attente
- ddlStatement - une seule instruction DDL qui effectue l'action requise

Remarques

Cette macro de fonction n'a aucun effet si l'utilisateur a choisi de générer du DDL dans un fichier.

EXIST_STRING ([ddlStatement])

Recherche dans la file d'attente d'exécution du Moteur d'Exécution l'instruction DDL fournie et renvoie 'T' si l'instruction est trouvée.

Paramètres

- ddlStatement - une seule instruction DDL

Remarques

Aucun.

GET_FIRST_SQL_KEYWORD ([statement])

Renvoie le premier mot-clé de l'instruction SQL fournie.

Paramètres

- instruction - l'instruction SQL

Remarques

Aucun.

ODBC_TABLE_TAGVALUE ([tagName])

Renvoie la valeur du nom de balise fourni dans la version de la base de données active du tableau actuel.

Paramètres

- tagName - le nom de l'élément de balise à récupérer

Remarques

Aucun.

PROCESS_DDL_SCRIPT ([type] , [parameter2] , [parameter3] , [parameter4])

Macro de fonction générique qui renvoie une string formatée dans un but spécifique.

Paramètres

- type - spécifie l'action spéciale à entreprendre
- paramètre2 - paramètre générique 2, aura un objectif différent pour chaque type
- paramètre3 - paramètre générique 3, aura un objectif différent pour chaque type
- paramètre4 - paramètre générique 4, aura un objectif différent pour chaque type

Remarques

Pour les synonymes Oracle, utilisez ces paramètres :

- tapez = "SYNONYMES"
- paramètre2 = le nom tableau ; par exemple, TBL_EMPLOYEES
- paramètre3 = une string de valeurs délimitée, séparées par des points-virgules, spécifiant le propriétaire et le nom du synonyme avec deux points entre eux ; par exemple, OE:EMPLOYES;PUBLIC:PUB_EMPLOYEES;
- paramètre4 = le terminateur de l'instruction

Retourner le résultat

Du format :

CRÉER LE SYNONYME OE.EMPLOYEES POUR TBL_EMPLOYEES ;

CRÉER LE SYNONYME PUBLIC PUB_EMPLOYEES POUR TBL_EMPLOYEES ;

REMOVE_LAST_SEPARATOR ([ddlStatement] , [separator])

Renvoie l'instruction DDL fournie avec le dernier séparateur supprimé (s'il existe).

Paramètres

- ddlStatement - une instruction DDL partielle
- séparateur - le caractère séparateur qui doit être supprimé

Remarques

Lors de la création d'une string représentant une instruction DDL, il est courant d'ajouter le caractère séparateur après chaque élément ; cependant, le séparateur n'est pas requis après le dernier élément, cette macro de fonction est donc fournie pour supprimer le séparateur de fin.

REMOVE_STRING ([ddlStatement])

Supprime l'instruction DDL fournie de la file d'attente d'exécution du Moteur d'Exécution .

Paramètres

- ddlStatement - une seule instruction DDL

Remarques

Aucun.

SUPPRESS_EXECUTE_CURRENT ([boolean])

Une macro de fonction pour activer/désactiver les appels ultérieurs à EXECUTE_CURRENT .

Paramètres

- booléen - Vrai ou Faux

Remarques

L'état par défaut de cet indicateur est False ; c'est-à-dire que les appels à EXECUTE_CURRENT ne sont pas ignorés.

Macros de propriétés DDL

Les macros Propriété DDL constituent un moyen pratique de récupérer les valeurs des propriétés d'un élément (c'est-à-dire Valeur Étiquetés). Dans le cadre de modélisation des données, il existe deux groupes de propriétés :

- Propriétés internes (celles qu'Enterprise Enterprise Architect reconnaît et utilise dans ses comparaisons) et
- Propriétés définies par l'utilisateur

Ces macros de propriétés donnent accès aux propriétés définies pour les différents éléments. Toutes les macros de propriétés ont la même syntaxe, renvoient une string et nécessitent que le nom de la propriété soit spécifié.

Syntaxe : propertyMacroName : "propertyName"

PROPRIÉTÉS INTERNES

tableBoolProperty :"propertyName"

Renvoie une représentation booléenne ("T" ou " ") de la valeur de la propriété interne dans la version du référentiel du Tableau actuel.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

tableProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version du référentiel du Tableau actuel.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

columnProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version du référentiel de la colonne actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

columnBoolProperty :"propertyName"

Renvoie une représentation booléenne ("T" ou " ") de la valeur de la propriété interne dans la version du référentiel de la colonne actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

constraintProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version du référentiel de la contrainte actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

constraintBoolProperty :"propertyName"

Renvoie une représentation booléenne ("T" ou " ") de la valeur de la propriété interne dans la version du référentiel de la contrainte actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

constraintColumnProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version du référentiel de la colonne de contrainte actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

constraintColumnBoolProperty :"propertyName"

Renvoie une représentation booléenne ("T" ou " ") de la valeur de la propriété interne dans la version du référentiel de la colonne de contrainte actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

viewProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version du référentiel de la Vue actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

procedureProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version du référentiel de la procédure actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

functionProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version du référentiel de la fonction actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

sequenceProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version du référentiel de la Séquence actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

packageProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version du référentiel de la base de données actuelle Paquetage .

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

odbcTableProperty :"propertyName"

Renvoie la valeur de la propriété interne dans la version ODBC du Tableau actuel.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

odbcConstraintProperty : "propertyName"

Renvoie la valeur de la propriété interne dans la version ODBC de la contrainte actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

PROPRIÉTÉS DÉFINIES PAR L'UTILISATEUR

tableUserProperty : "propertyName"

Renvoie la valeur de la propriété définie par l'utilisateur dans la version du référentiel du Tableau actuel.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

columnUserProperty : "propertyName"

Renvoie la valeur de la propriété définie par l'utilisateur dans la version du référentiel de la colonne actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

constraintUserProperty : "propertyName"

Renvoie la valeur de la propriété définie par l'utilisateur dans la version du référentiel de la contrainte actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

constraintColumnUserProperty : "propertyName"

Renvoie la valeur de la propriété définie par l'utilisateur dans la version du référentiel de la colonne de contrainte actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

viewUserProperty : "propertyName"

Renvoie la valeur de la propriété définie par l'utilisateur dans la version actuelle du référentiel Vue .

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

procedureUserProperty : "propertyName"

Renvoie la valeur de la propriété définie par l'utilisateur dans la version du référentiel de la procédure actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

functionUserProperty : "propertyName"

Renvoie la valeur de la propriété définie par l'utilisateur dans la version du référentiel de la fonction actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

sequenceUserProperty : "propertyName"

Renvoie la valeur de la propriété définie par l'utilisateur dans la version du référentiel de la Séquence actuelle.

Paramètres

- propertyName - le nom de la propriété à récupérer

Remarques

Aucun.

Options DDL dans Gabarits

Les macros Options de génération DDL offrent aux gabarits DDL un moyen pratique d'accéder aux options de génération.

Cette liste identifie et décrit brièvement chacune des macros d'options disponibles. Chaque option a une valeur de « T » pour vrai ou une string vide pour faux.

ddlGenerateToExecuteEngine

Dirige le DDL généré vers le Moteur d'Exécution .

ddlOptionColumnComments

Incluez les commentaires de colonne dans le DDL généré.

ddlOptionGenerateCheck

Incluez les contraintes Check dans le DDL généré.

ddlOptionGenerateDrop

Incluez les instructions DROP dans le DDL généré.

ddlOptionGenerateForeign

Incluez Foreign Keys dans le DDL généré.

ddlOptionGenerateFunction

Incluez des fonctions dans le DDL généré.

ddlOptionGenerateIndex

Incluez les index dans le DDL généré.

ddlOptionGenerateLengthSemantic

(Oracle uniquement) Incluez la syntaxe sémantique de longueur dans les colonnes de texte dans le DDL généré.

ddlOptionGenerateNullable

Incluez le mot-clé NULL dans chaque colonne si elle n'a pas été marquée comme colonne NOT NULL dans le DDL généré.

ddlOptionGeneratePackage

(Oracle uniquement) Incluez Paquetages dans le DDL généré.

ddlOptionGeneratePrimary

Incluez les contraintes Primary Key dans le DDL généré.

ddlOptionGenerateProcedure

Incluez les procédures dans le DDL généré.

ddlOptionGenerateSeparateConstraint

Générer des contraintes Tableau séparément de l'instruction CREATE TABLE ; c'est-à-dire en utilisant une instruction ALTER TABLE.

Note : Certains SGBD ne prennent pas support des contraintes distinctes dans toutes les conditions.

ddlOptionGenerateSequence

Incluez les séquences dans le DDL généré.

ddlOptionGenerateTable

Incluez Tableaux dans le DDL généré.

ddlOptionGenerateTableProperty

Incluez les propriétés étendues sur Tableaux dans le DDL généré.

ddlOptionGenerateTrigger

Incluez Tableau Déclencheurs dans le DDL généré.

ddlOptionGenerateUnique

Incluez des contraintes uniques dans le DDL généré.

ddlOptionGenerateView

Incluez Vues dans le DDL généré.

ddlOptionHeaderComments

Incluez les commentaires d'en-tête dans le DDL généré.

ddlOptionTableComments

Incluez les commentaires Tableau dans le DDL généré.

ddlOptionUseAlias

Utilisez des alias au lieu de noms pour tous les objets (composants object) comme spécifié sur l'écran Générer DDL.

ddlOptionUseDatabaseName

Incluez l'instruction USE DATABASE au début de chaque fichier généré.

ddlUseAlias

Utilisez des alias au lieu de noms pour tous les objets (composants object) comme spécifié dans l'onglet « Comparaison de bases de données » du Générateur de bases de données.

Limites du DDL

Une fonctionnalité fondamentale d'un système de gestion de base de données (SGBD) est de permettre la définition d'objets de base de données via un langage structuré ; ce langage est appelé DDL (pour Data Definition Language, ou langage de description de données). La syntaxe DDL de chaque SGBD est unique. Bien qu'il existe des instructions DDL et des mots-clés communs à tous les SGBD, il existe des différences qui nécessitent que chaque SGBD ait son propre ensemble de gabarits DDL dans Enterprise Architect .

Cette page résume les principales limitations de chacun des systèmes de gestion de base de données pris en charge.

Accès MS

- Les commentaires ne peuvent pas être appliqués (ou modifiés) Tableaux , aux Colonnes Tableau , aux Contraintes Tableau ou Vues . Par conséquent, Enterprise Architect ignore ces différences.
- L'instruction CREATE TABLE ne prend pas support la définition des valeurs par défaut des colonnes. Par conséquent, Enterprise Architect exclut la définition par défaut de tous les DDL générés ; cependant, cela met en évidence une différence par défaut dans la logique de comparaison
- Généralement, les noms object en DDL peuvent être placés entre crochets ([ ]) afin qu'ils puissent inclure des espaces et d'autres caractères non standard. Cependant, l'instruction CREATE VIEW DDL ne prend pas support la notation entre crochets ; le gabarit DDL 'Create Vue ' remplace tous les espaces par des caractères de soulignement ('_')

MySQL

- Les commentaires ne peuvent être appliqués aux index et aux contraintes uniques que lorsque la version de MySQL est supérieure à 5.5.3.
- Les commentaires ne peuvent être appliqués aux index et aux contraintes uniques qu'au moment de leur création. Par conséquent, la modification du commentaire d'un index ou d'une contrainte unique entraîne la suppression et la recréation de la contrainte.
- Les contraintes de vérification ne sont pas prises en charge ; alors que le moteur MySQL DDL peut analyser de telles déclarations, il les ignore simplement
- Les commentaires ne peuvent pas être appliqués (ou modifiés) aux Vues , procédures ou fonctions. Enterprise Architect ignore donc ces différences.

Oracle

- Les commentaires ne peuvent pas être appliqués (ou modifiés) aux procédures, séquences ou fonctions. Enterprise Architect ignore donc ces différences.

PostgreSQL

- Actuellement, Enterprise Architect ne prend pas support les paramètres de fonction. Par conséquent, toute instruction (COMMENT ON ou DROP) faisant référence à une fonction par son nom échouera car elle doit utiliser une combinaison de nom de fonction et de paramètres.

SQL Lite

- Les contraintes ne peuvent pas être ajoutées à un Tableau existant ; le Tableau doit être supprimé et créé (y compris la nouvelle contrainte dans l'instruction Create)
- Les commentaires ne sont pris en charge sur aucun type object , par conséquent Enterprise Architect ignore toutes les différences de remarques

Importer le schéma de base de données

La puissance de l'ingénierie basée sur des modèles réside dans la capacité de visualiser, d'analyser et de concevoir tous les aspects d'un système. Être capable de visualiser les schémas de base de données aux côtés d'autres modèles d'un système offre une grande clarté et réduit le risque d'erreur. Enterprise Architect peut procéder à l'ingénierie inverse d'un schéma de SGBD et de ses objets dans un modèle selon un certain nombre de normes différentes, notamment UML , Information Engineering et IDEF 1X. Un large éventail d'objets de base de données sont pris en charge, notamment Tableaux , Vues , les Procédures, les Fonctions et les Séquences. Enterprise Architect y parvient en interrogeant le schéma d'information du SGBD et en important la définition dans un objet UML . Au fur et à mesure que des modifications sont apportées à la base de données Live, les modifications peuvent être synchronisées dans le modèle.

Une fois le schéma dans Enterprise Architect , les objets de la base de données peuvent être retracés jusqu'à d'autres éléments, garantissant ainsi l'intégrité de la conception et de l'architecture. Lorsque les systèmes ciblent plusieurs SGBD, ceux-ci peuvent tous faire l'objet d'une ingénierie inverse dans un modèle et les éléments et types de données peuvent être comparés entre ces modèles. Le moteur de reporting sophistiqué peut produire une documentation de haute qualité, comprenant des dictionnaires de données, diagrammes et des relations avec d'autres modèles tels que l'architecture et les exigences en matière d'informations, et finalement avec les objectifs et les moteurs de l'entreprise.

Les informations sur le schéma de base de données peuvent être importées via le générateur de base de données (recommandé) ou à partir du ruban « Développer ».

Importer le schéma de base de données

Étape	Action
1	Ouvrez le Database Builder (Développement > Modélisation de données > Database Builder)
2	Chargez ou créez un Data Modèle .
3	• Cliquez-droit sur le Data Modèle chargé dans le Database Builder et sélectionnez 'Import DB schema' ou • Dans le ruban, sélectionnez "Développer > Modélisation de données > Importer". La dialogue « Importer un schéma de base de données » s'affiche, affichant les détails de la connexion à la base de données active actuelle.

La dialogue Importer un schéma de base de données

Option	Description
Base de données	Ce champ affiche une description de la connexion Live actuelle, au format : dbms.database_server.database_name Si nécessaire, cliquez sur le bouton et sélectionnez une connexion alternative.
Importer vers	Ce champ affiche le Paquetage cible dans lequel les nouveaux objets seront enregistrés. Si vous souhaitez spécifier un Paquetage différent, cliquez sur le bouton et sélectionnez un Paquetage alternatif.

Inclure uniquement les objets des schémas	Si le type de base de données supporte plusieurs schémas (tels que SQL Server, Oracle, PostgreSQL et DB2 Express), vous pouvez filtrer les objets à récupérer de la base de données par schéma. Les schémas disponibles sont automatiquement répertoriés dans ce panneau. Cochez la case en regard de chaque schéma à inclure dans l'importation. (Vous pouvez cliquer sur le bouton Tous pour sélectionner tous les schémas ou sur le bouton Aucun pour décocher toutes les cases sélectionnées.) Si vous pensez que la liste des schémas a pu changer depuis que vous les avez chargés, vous pouvez actualiser la liste en cliquant sur le bouton Recharger les schémas.
Filtre de nom	Le champ 'Name Filter:' permet de filtrer les objets à l'aide de caractères génériques SQL appropriés au SGBD du schéma en cours d'importation. Par exemple, pour Oracle : • LIKE 'A%' - liste les objets dont le nom commence par la lettre 'A' • NOT LIKE '%_%' ESCAPE '\' - liste les objets dont le nom n'inclut pas de trait de soulignement (_) • IN ('TABLE1','TABLE2') - liste les objets dont les noms sont inclus entre parenthèses • NOT IN ('TABLE1','TABLE2') - liste les objets dont les noms ne sont pas inclus entre parenthèses Note qu'un seul filtre peut être saisi. Vous ne pouvez pas ajouter un deuxième filtre à l'aide de la clause AND . Le filtrage n'est pas disponible pour MS Access
Options de filtre	Le panneau « Options de filtre » contrôle les types object et les propriétés qui sont lus à partir du schéma de base de données. Les valeurs modifiées sur cet écran sont enregistrées dans le registre afin d'être réappliquées lors de la prochaine session de travail. Les options disponibles sont brièvement décrites ici ; cochez la case en regard d'une option pour l'activer. **Tableaux** • Tableaux - Sélectionnez pour importer Tableaux • Primary Keys Tableau : sélectionnez cette option pour importer des définitions Primary Key sur Tableaux • Foreign Keys Tableau : sélectionnez cette option pour importer des définitions Foreign Key sur Tableaux • Index Tableau - Sélectionnez pour importer des index Tableau • Contraintes uniques : sélectionnez cette option pour importer des définitions de contraintes uniques sur Tableaux • Contraintes de vérification - Sélectionnez cette option pour importer les définitions de contraintes de vérification sur Tableaux • Tableau Déclencheurs - Sélectionnez pour importer les définitions Déclencheur sur Tableaux • Propriétés Tableau : sélectionnez cette option pour importer les propriétés Tableau étendues. • Propriétés de contrainte : sélectionnez cette option pour importer Propriétés de contrainte pour Tableaux • Sémantique de longueur : sélectionnez cette option pour importer des définitions sémantiques de longueur sur les colonnes string Oracle. **Objets** • Vues - Sélectionnez pour importer Vues

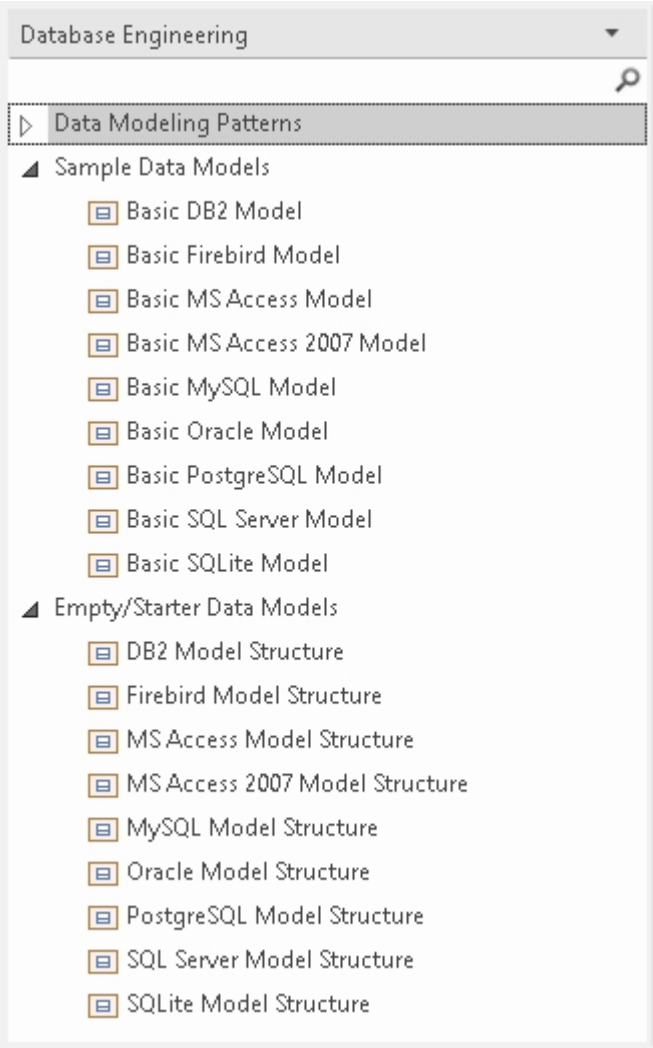
	• Procédures - Sélectionnez pour importer des procédures - En tant qu'opérations - Sélectionnez cette option pour importer des procédures en tant qu'opérations (méthodes) d'une seule classe ; tu peux affichez-les et modifiez-les via la boîte dialogue ' Propriétés ' du conteneur object de base de données (l'option par défaut à non sélectionné, où les éléments sélectionnés sont importés en tant que classes distinctes) • Fonctions - Sélectionnez pour importer des fonctions - En tant qu'opérations - Sélectionnez pour importer des fonctions en tant qu'opérations (par défaut, non sélectionné) • Séquences - Sélectionnez pour importer des séquences - En tant qu'opérations - Sélectionnez pour importer des séquences en tant qu'opérations (par défaut, non sélectionné) • Paquetage - Sélectionnez pour importer Paquetages Oracle **Avancé** • Objets système - Sélectionnez pour importer Tableaux système, Vues et d'autres objets système. **Attention** : Avec l'option 'En tant qu'opérations' pour les procédures, fonctions et séquences, si des objets ont été importés sous un paramètre (sélectionné ou non) et que vous modifiez ensuite le paramètre et importez d'autres objets, les objets importés sous le premier paramètre sont supprimés.
Synchronisation	Sélectionnez le bouton radio approprié pour indiquer si les classes existantes doivent être mises à jour ou les objets de base de données importés en tant que nouveaux objets. Si vous sélectionnez l'option « Synchroniser les classes existantes », cochez également les cases appropriées pour déterminer si les commentaires du modèle, les valeurs par défaut des colonnes et/ou les contraintes Tableau doivent être conservés ou écrasés par les commentaires, les valeurs et les contraintes des objets importés.
Importer vers	Sélectionnez le bouton radio approprié pour indiquer s'il faut mettre à jour le Paquetage et diagrammes du modèle de données actuellement ouverts , ou simplement le Paquetage . Si aucun diagrammes n'est ouvert, le bouton radio « Paquetage uniquement » est sélectionné par défaut et les options sont désactivées ; si les diagrammes ouverts se trouvent dans le Paquetage sélectionné, vous pouvez sélectionner l'une ou l'autre option.
Importer	Cliquez sur ce bouton pour lancer l'importation. La dialogue « Sélectionner les objets de base de données à importer » s'affiche, répertoriant tous les objets de base de données trouvés qui correspondent aux critères de sélection. Cochez la case en regard de chaque schéma (ou type object) pour sélectionner automatiquement tous les objets de ce groupe ou pour importer chaque object individuellement. Cliquez sur le bouton Tous pour sélectionner tous les types et objets, ou sur le bouton Aucun pour décocher toutes les cases sélectionnées. Lorsque vous avez sélectionné tous les objets à importer, cliquez sur le bouton OK pour continuer l'importation.

Notes

- Dans Windows , le DSN ODBC peut être défini pour les applications 32 ou 64 bits. Il faut donc veiller à ce que tous les DSN ODBC destinés à l'utilisation Enterprise Architect soient définis en partageant la même architecture. Ceci est particulièrement important à partir de la version 16 Enterprise Architect car il est maintenant disponible en versions 32 et 64 bits. Une solution alternative (et recommandée par Sparx Systems) consiste à utiliser des connexions natives, car elles fonctionnent pour les deux architectures.
- La connexion ODBC doit utiliser le pilote ODBC disponible auprès du fournisseur de SGBD, tel que le pilote ODBC de MySQL pour MySQL et le pilote ODBC d'Oracle pour Oracle ; les pilotes fournis par des fournisseurs tiers ne sont pas pris en charge, y compris le pilote Microsoft ODBC pour Oracle
- Vous pouvez importer un pilote ODBC approprié pour SQLite depuis http://www.ch-werner.de/sqliteodbc/
- En raison des limitations de SQLite, round retour entre SQLite Tableau et les commentaires de colonnes n'est pas possible ; pour conserver les commentaires saisis dans un modèle de données SQLite lors de l'importation depuis ODBC, décochez la case « Remplacer les commentaires Object » dans la section « Synchronisation » de la dialogue de dialogue « Importer un schéma de base de données à partir d'une source ODBC ».
- Si vous configurez une connexion ODBC pour l'ingénierie inverse, les paramètres par défaut sont suffisants
- La liste des types de données Modélisation de données est définie comme des données statiques (dans chaque référentiel), donc en fonction de l'âge de votre référentiel, des types de données supplémentaires peuvent être disponibles dans la section « Types de données Modélisation de données » de la page « Ressources » sur le site Web de Sparx Systems

Création et gestion de modèles de données

Enterprise Architect est une plateforme modélisation de bases de données complète qui permet à l'utilisateur de travailler avec ses modèles de données physiques à toutes les étapes, de la conception jusqu'à la mise en œuvre de la base de données active, pour une large gamme de systèmes de gestion de bases de données tels que Microsoft SQL Server, Oracle, PostgreSQL et MySQL.



Cette figure montre les motifs assistant du modèle de démarrage pour la conception de bases de données pour une gamme de SGBDR.

Paramètres Modélisation des données

Enterprise Architect fournit des paramètres modélisation de données qui peuvent être utilisés pour configurer la manière dont les systèmes de base de données sont modélisés dans Enterprise Architect . Ceux-ci incluent la possibilité de définir le langage modélisation de données, qui détermine la manière dont les connecteurs sont affichés, ainsi que les paramètres permettant de configurer la dénomination des Primary Keys , Foreign Keys et des index. Les paramètres sont globaux et affecteront n'importe quel référentiel Enterprise Architect .

Accéder

Ruban	Démarrer > Apparence > Préférences > Préférences > Ingénierie du code source > Éditeurs de Code > DDL

Éditeur DDL

Dans ce champ, vous recherchez le chemin complet du fichier d'exécution et le nom d'un programme externe qu'Enterprise Enterprise Architect doit utiliser pour ouvrir les fichiers créés par sa fonctionnalité Générer DDL. Si vous laissez ce champ vide, Enterprise Architect utilise l'éditeur de code par défaut.

Base de données par défaut

Dans ce champ, vous sélectionnez le SGBD qui sera automatiquement affecté aux objets de base de données créés en dehors d'un espace de travail Data Modèle (voir la rubrique d'aide *Créer un Data Modèle à partir de Modèle Motif*).

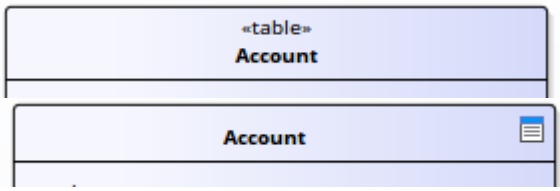
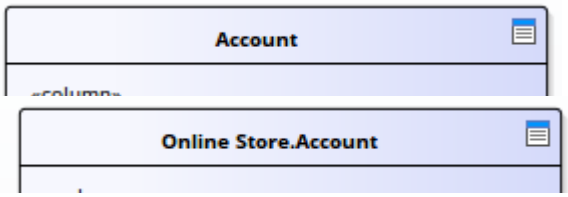
Stockage MySQL

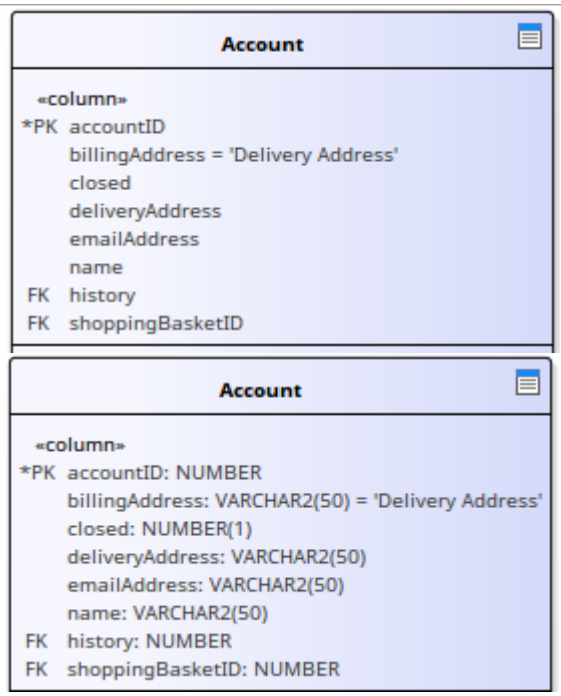
Dans ce champ vous sélectionnez le moteur de stockage par défaut à attribuer aux Tableaux MySQL ; à partir de MySQL v 5.5, la valeur par défaut est InnoDB.

Notations Modélisation des données

Enterprise Architect supporte de nombreux paramètres liés à modélisation des données qui peuvent influencer la façon dont les objets de base de données sont représentés sur diagrammes . Ces paramètres et la manière dont ils peuvent affecter la représentation des objets de base de données sont décrits ici.

Paramètres

Paramètre	Détail
Icônes de stéréotypes	Accès : 'Conception > Diagramme > Gérer > Propriétés > Éléments : Utiliser les icônes de stéréotype' Valeur par défaut : Vrai Enterprise Architect fournit un paramètre au niveau du diagramme pour l'affichage des objets stéréotypés. Lorsque la case est cochée, les objets de base de données sur le diagramme sont affichés avec une icône représentant leur stéréotype au lieu du nom du stéréotype. 
Afficher le propriétaire Modèle données	Accès : 'Conception > Diagramme > Gérer > Propriétés > Eléments : Afficher le propriétaire Modèle données' Valeur par défaut : Vrai Le système fournit un paramètre au niveau du diagramme pour l'affichage des propriétaires. Lorsque la case est cochée, les objets de base de données sur le diagramme actuel seront affichés avec leur nom complet « {Owner.}ObjectName ». 
Afficher les détails de la colonne	Accès : 'Conception > Diagramme > Gérer > Propriétés > Fonctionnalités : Afficher le détail des attributs' Valeur par défaut : Nom uniquement Le système fournit un paramètre au niveau du diagramme pour l'affichage des noms de colonnes et des types de données Tableau . Les options disponibles sont : 'Nom uniquement' ou 'Nom et Type '.



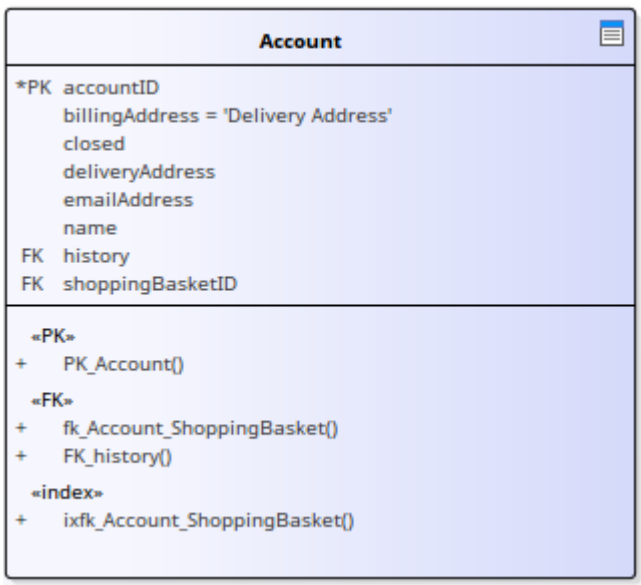
Afficher les détails de la colonne impliquée

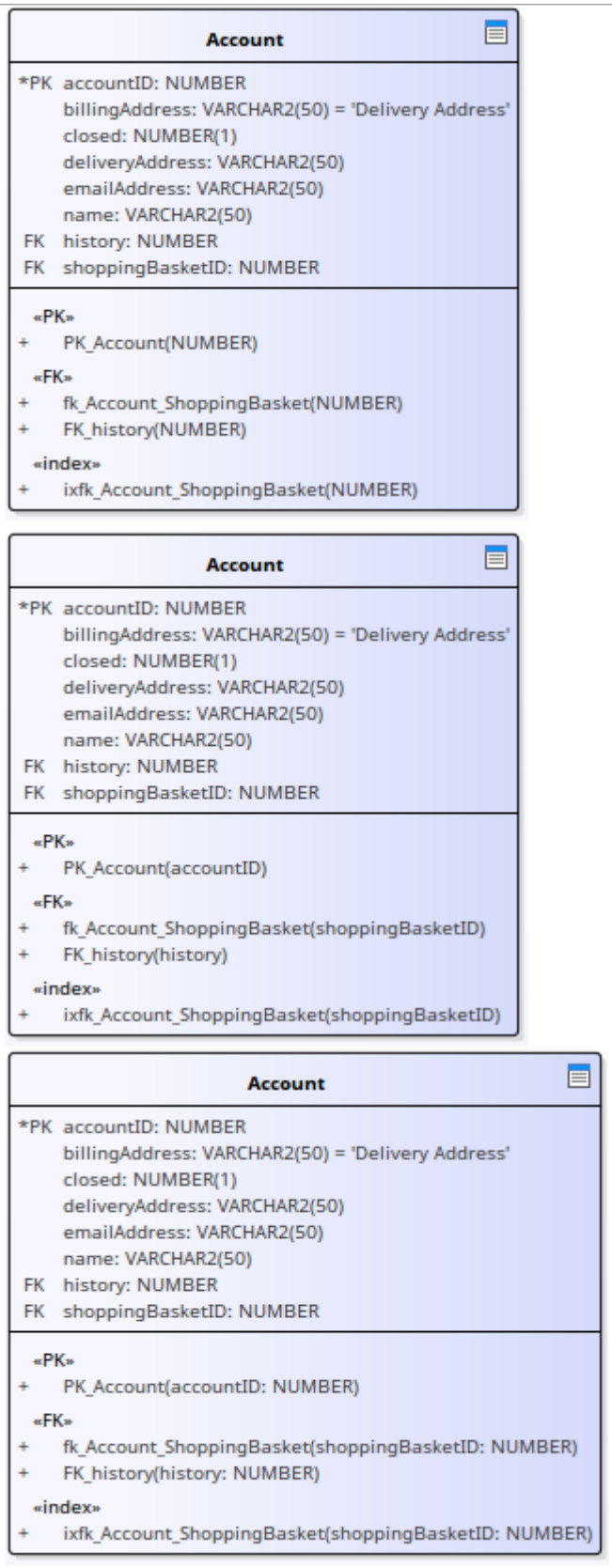
Accès : 'Conception > Diagramme > Gérer > Propriétés > Fonctionnalités | Afficher les détails des paramètres'

Valeur par défaut : Type uniquement

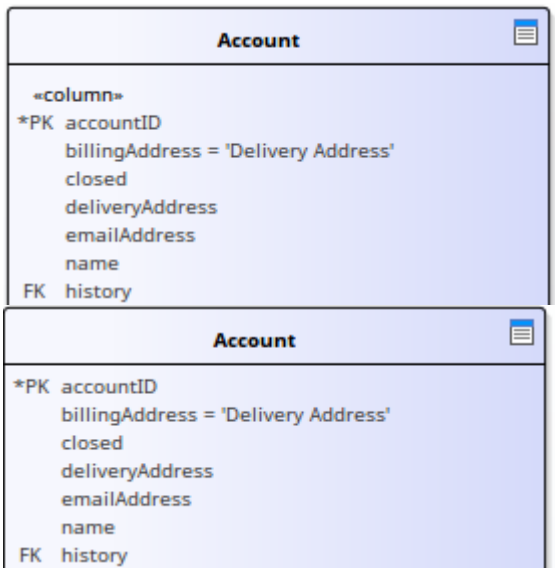
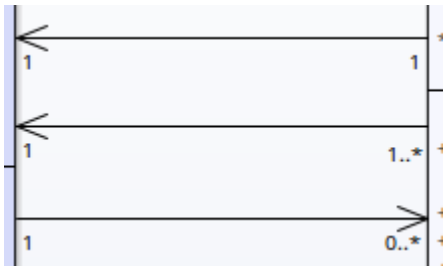
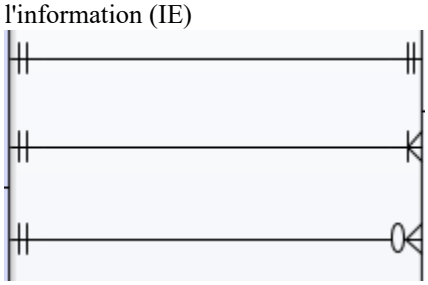
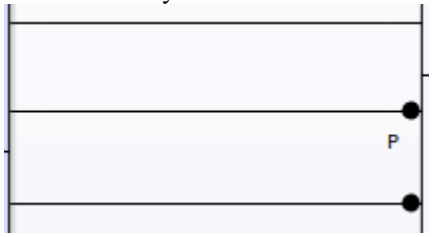
Le système fournit un paramètre au niveau du diagramme pour l'affichage des colonnes impliquées dans une contrainte Tableau . Les options disponibles sont : "Aucun", " Type uniquement", "Nom uniquement" et "Détails complets".

Dans ces exemples, la contrainte Primary Key (PK) « PK_account » implique la colonne « accountID ».





Afficher le stéréotype de la colonne	Accès : Démarrer > Application > Préférences > Préférences > Objets : Afficher le stéréotype « colonne » Valeur par défaut : Vrai Enterprise Architect fournit un paramètre de niveau global qui contrôle si le

	stéréotype « colonne » est affiché ou non au-dessus des colonnes de chaque Tableau . Vous pouvez donc masquer le stéréotype si vous préférez, sachant que les attributs avec un stéréotype de « colonne » sont la seule option valable pour Tableaux . 
Notation du connecteur	Accès : 'Conception > Diagramme > Gérer > Propriétés > Connecteurs : Notation du connecteur' Valeur par défaut : UML 2.1 Enterprise Architect supporte trois notations diagramme pour modélisation des données : • UML 2.1 - la notation UML 2.1 standard pour les connecteurs  • Ingénierie de l'information - le style de connexion de l'ingénierie de l'information (IE)  • IDEF1X - le style de connexion IDEF1X Méthodes de définition intégrées  (Ce sont les trois mêmes connecteurs utilisant des notations différentes.) La notation par défaut pour le diagramme Modélisation de données est « Information

	Engineering », tandis que la notation par défaut pour les modèles créés à partir Ingénierie Base de Données Motifs est « IDEF1X ».

Nom DDL Gabarits

À différents moments du processus de modélisation des données, Enterprise Architect doit générer automatiquement des contraintes Tableau . La norme de dénomination de ces contraintes générées est définie et appliquée par le DDL Name Gabarits , que vous êtes libre de modifier à tout moment. Ces gabarits de nom sont définis au niveau du référentiel, donc chaque fois qu'ils seront modifiés, tous les utilisateurs du référentiel utiliseront les nouveaux gabarits .

Accéder

Ruban	Paramètres > Modèle > Options > Ingénierie du code source : Gabarits du nom DDL

Nom DDL Gabarits

Option	Action
Primary Key	Définissez le nom gabarit utilisé lors de la création de contraintes Primary Key .
Contrainte unique	Définissez le nom gabarit utilisé lors de la création de contraintes uniques.
Foreign Key	Définissez le nom gabarit utilisé lors de la création de contraintes Foreign Key .
Index Foreign Key	Définissez le nom gabarit utilisé lors de la création des index Foreign Key .
Sauvegarder	Cliquez sur ce bouton pour enregistrer le(s) nom(s) gabarit (s) que vous avez défini.

Macros Gabarit

Ces macros reconnues seront remplacées par un texte de nom lors de la création d'un nom de contrainte.

Macro	S'applique à
%nom de la table%	Primary Key Contrainte unique Description : string remplacée par le nom du Tableau .
%nom de colonne%	string remplacée par le(s) nom(s) de colonne de la contrainte.
%nom de la table primaire%	Foreign Key Description : string remplacée par le nom du Tableau principal (parent).
%nom de table étrangère%	string qui est remplacée par le nom du Tableau étranger (enfant).

%nom de clé étrangère%	Index Foreign Key Description : string remplacée par le nom de la Foreign Key .

Types de données

Chaque colonne Tableau que vous définissez dans votre modèle de données est associée à un type de données qui spécifie le type d'informations pouvant être stocké par la colonne. Les types de données disponibles pour une colonne dépendent du SGBD sélectionné pour Tableau , car chaque SGBD supporte sa propre liste de types de données. Bien que chaque SGBD supporte les mêmes types de base, tels que string , les nombres entiers ou décimaux, chaque SGBD les appelle sous des noms différents et possède des propriétés différentes.

Chaque référentiel Enterprise Architect contient les définitions des types de données de base pour un certain nombre de produits SGBD standard. Cependant, comme les types de données varient d'un produit SGBD à un autre, et d'une version d'un produit à l'autre, Enterprise Architect vous propose des outils pour :

- Définir de nouveaux types de données pour une nouvelle version de votre produit SGBD
- Définir des types de données pour un nouveau produit de base de données non standard
- Convertir automatiquement les types de données d'un produit SGBD défini à un autre
- Importer et exporter des types de données entre référentiels

Mapper les types de données entre les produits SGBD

Bien que modélisation des modèles de données physiques fournisse beaucoup de détails sur tous Tableaux et leurs colonnes, ce niveau de détail rend plus difficile le changement de technologie ou de plate-forme cible. Par exemple, après avoir procédé à l'ingénierie inverse de votre base de données en un modèle de données physique, vous devez remapper les types de données avant de générer le schéma du nouveau produit SGBD.

Enterprise Architect fournit un ensemble de mappages par défaut pour les produits SGBD standard pris en charge, pour vous aider à automatiser le processus de conversion.

Toutefois, vous souhaiterez peut-être personnaliser les mappages par défaut en fonction des exigences spécifiques de votre projet ou lorsque le mappage d'un type de données à un autre n'est pas actuellement défini. Par exemple, lors de votre migration d'une plate-forme SGBD à une autre, l'une des plates-formes peut être non standard ou non prise en charge par Enterprise Architect .

Accéder

Ruban	Paramètres > Données de référence > Paramètres > Types de données de base de données : Mappage des types de données

Mappage des types de données de base de données

Répétez ce processus pour chacun des types de données à mapper.

Une fois que vous êtes satisfait des mappages de types de données, vous pouvez convertir Tableaux individuels ou un Paquetage entier de Tableaux vers le nouveau produit SGBD cible.

Champ/Bouton	Action
Du nom du produit	Cliquez sur la flèche déroulante et sélectionnez le produit SGBD à partir duquel mapper les types de données.
Types de données définis pour les bases de données	Affiche tous les types de données définis pour le produit et, le cas échéant, leurs tailles et valeurs. Cliquez sur le type de données à mapper - celui-ci doit avoir une unité de taille et valeur définies. Les champs « Type de données » et « Type commun » sous le champ « Nom du produit » affichent ce type de données.
Vers le nom du produit	Cliquez sur la flèche déroulante et sélectionnez le produit SGBD auquel mapper les types de données. Les champs « Type de données » et « Type commun » sous ce champ affichent les valeurs correspondant à celles des champs du produit « De ».
Taille	Cliquez sur le bouton radio de l'unité de taille appropriée et saisissez les valeurs par défaut dans les champs de données correspondants.
Sauvegarder	Cliquez sur ce bouton pour enregistrer le mappage.

Conversion de produit SGBD pour un Paquetage

À l'aide du mappeur DBMS Paquetage , vous pouvez convertir automatiquement un Paquetage de Tableaux de base de données d'un type de SGBD pris en charge vers un autre type de SGBD pris en charge. Vous pouvez également modifier le type de SGBD pour Tableaux individuels.

Si l'un des types de SGBD n'est pas standard ou n'est pas pris en charge par Enterprise Architect , vous devez vérifier que le mappage des types de données d'un type de SGBD à l'autre a été défini.

Accéder

Ruban	Conception > Paquetage > Gérer > Options du SGBD Développer > Modélisation de données > Générateur de base de données > Cliquez-droit sur la base de données souhaitée | Charger | cliquez-droit sur le nœud racine | Gérer les options du SGBD

Mappez les types de données SGBD d'un Paquetage aux types de données d'un autre SGBD

Champ/Bouton	Action
Inclure des objets dans Paquetages enfants	S'il y a des objets dans Paquetages enfants qui doivent également être modifiés, cochez la case.
Changer de SGBD	Cochez la case.
SGBD actuel	Cliquez sur la flèche déroulante et sélectionnez le SGBD actuel.
Nouveau SGBD	Cliquez sur la flèche déroulante et sélectionnez le SGBD cible.
OK	Cliquez sur ce bouton pour mapper tous Tableaux Paquetages sélectionnés vers le nouveau SGBD.

Conversion Type de données pour un Tableau

Une fois qu'un schéma de base de données a été configuré sur un diagramme Enterprise Architect (soit en important via ODBC, soit en configurant manuellement les Tableaux), le SGBD peut être remplacé par un autre type et les types de données de colonnes sont mappés en conséquence pour chaque Tableau .

Vous pouvez utiliser cette procédure si vous avez copié un petit nombre de Tableaux dans le projet depuis ailleurs, mais si vous avez plusieurs Tableaux , vous pouvez également les convertir tous en même temps dans leur Paquetage parent.

Si l'un des types de SGBD n'est pas standard ou n'est pas pris en charge par Enterprise Architect , vous devez vérifier que le mappage des types de données d'un type de SGBD à l'autre a été défini.

Mapper le type de SGBD d'un Tableau à un autre type de SGBD

Étape	Action
1	Double-cliquez sur l'élément Tableau dans un diagramme . La dialogue Tableau « Propriétés » s'affiche, avec le champ « Base de données » affichant le SGBD actuel pour ce Tableau .
2	Pour mapper les types de données à un autre SGBD, cliquez sur la flèche déroulante « Base de données » et sélectionnez le SGBD cible.
3	Cliquez sur le bouton Appliquer. Les types de données sont convertis pour correspondre à ceux du nouveau SGBD, et ceux-ci sont reflétés dans tout DDL généré à partir de ce Tableau .

Types de données de base de données

À l'aide de dialogue « Types de données de base de données » d' Enterprise Architect , vous pouvez ajouter à l'ensemble des types de données disponibles pour un SGBD particulier. Tu peux:

- Identifiez le SGBD utilisé et, si nécessaire, définissez-le comme modèle par défaut
- Incluez tous les nouveaux types de données pris en charge par les versions ultérieures du SGBD et non encore inclus dans Enterprise Architect
- Supprimez tous les types de données précédemment ajoutés qui ne sont plus pertinents
- Ajoutez un nouveau produit SGBD et ses types de données intégrés si, par exemple, vous souhaitez créer un modèle de données physique pour un produit SGBD qui n'est pas encore pris en charge nativement par Enterprise Architect .

Accéder

Ruban	Paramètres > Données de référence > Paramètres > Types de données de base de données ou Développement > Modélisation des données > Types de données

Gérer les types de données

Vous pouvez transporter ces types de données de base de données entre les modèles Enterprise Architect à l'aide des options « Exporter les données de référence » et « Importer les données de référence ».

Champ/Bouton	Action
Nom du produit	Cliquez sur la flèche déroulante et sélectionnez un SGBD existant. Une fois qu'un produit est sélectionné, tous les types de données définis seront affichés dans la liste « Types de données définis pour les bases de données ».
Ajouter un produit	Si votre SGBD n'est pas répertorié, cliquez sur ce bouton pour l'ajouter. Une invite « prompt » s'affiche, dans laquelle vous saisissez le nom du SGBD ; cliquez sur le bouton OK pour ajouter le nom à la liste déroulante.
Définir par défaut	Cochez la case pour définir le SGBD sélectionné comme système par défaut pour l'ingénierie et modélisation de votre base de données. Une fois que vous avez défini la base de données par défaut, lorsque vous créez de nouveaux éléments Tableau , le type de base de données est automatiquement prédéfini sur cette valeur par défaut. Vous pouvez également définir le type de base de données par défaut dans le deuxième champ de saisie de données de la barre d'outils Génération de code.
Nouveau	Cliquez sur ce bouton pour effacer les champs de type de données sur le dialogue afin de pouvoir définir un autre type de données.
Type de données	Type un nom pour le type de données.
Taille	Sélectionnez le bouton radio approprié pour la taille requise et, le cas échéant,

	spécifiez les valeurs par défaut et maximales : • Aucun – pour les types de données sans composant de taille, tels que INT • Longueur – pour les types de données qui nécessitent une taille unique définissant la longueur, comme VARCHAR(10) • Précision et échelle – pour les types de données nécessitant deux valeurs numériques, telles que DECIMAL(18,2)
Type commun	Cliquez sur la flèche déroulante et sélectionnez le nom générique de chaque type de données. Ceci est utilisé lorsque le SGBD d'un Tableau est modifié.
Sauvegarder	Cliquez sur le bouton pour enregistrer immédiatement votre type de données dans le référentiel (et l'ajouter à la liste « Types de données définis pour les bases de données »).
Types de données définis pour les bases de données	Ce panneau répertorie les types de données actuellement définis pour le SGBD sélectionné, fournis par le système ou définis par l'utilisateur.
Supprimer	Sélectionnez un type de données dans la liste « Types de données définis pour les bases de données » et cliquez sur ce bouton pour supprimer le type de données.
Carte des types de données	Si vous avez modifié le SGBD ou la technologie pour laquelle vous avez défini les types de données depuis ou vers un type de SGBD non pris en charge, cliquez sur ce bouton pour définir comment remapper automatiquement les types de données vers votre nouveau SGBD ou technologie.

Types de données MySQL

MySQL supporte les types de données ENUM et SET, qui doivent être ajoutés à votre modèle Enterprise Architect avant de pouvoir les utiliser comme types de colonnes.

Accéder

Ruban	Paramètres > Données de référence > Paramètres > Types de données de base de données

Ajoutez les types de données ENUM et SET pour MySQL

Lorsque vous utilisez ces types de données ultérieurement dans le champ « Initial » d'une colonne, saisissez les valeurs sous forme de liste séparée par des virgules, au format :

('un deux trois')

Si une valeur est la valeur par défaut, utilisez le format :

("un", "deux", "trois") par défaut "trois"

Étape	Action
1	La dialogue « Types de données de base de données » s'affiche.
2	Dans le champ « Nom du produit », sélectionnez « MySQL ».
3	Ajoutez les types de données ENUM et SET.

Types de données Oracle

Les types de données Oracle NUMBER et VARCHAR possèdent des propriétés supplémentaires que vous pouvez modéliser.

Accéder

Ruban	Paramètres > Données de référence > Paramètres > Types de données de base de données

Types de données

Type de données	Détail
NOMBRE	Le type de données NUMBER nécessite des propriétés de précision et d'échelle. Les champs 'Précision' et 'Echelle' sont affichés sur la page ' Attributes ' de la fenêtre Fonctionnalités lorsque le type de données est défini sur NOMBRE ; si vous saisissez des informations dans ces champs, elles s'affichent sur vos diagrammes . Par exemple: créez un NOMBRE en définissant 'Precision' = 0 et 'Scale' = 0 créez NUMBER(8) en définissant 'Precision' = 8 et 'Scale' = 0 créez NUMBER(8,2) en définissant 'Precision' = 8 et 'Scale' = 2
VARCHAR	Les types de données Oracle VARCHAR2(15 CHAR) et VARCHAR2(50 BYTE) peuvent être créés en ajoutant la Valeur Étiquetée LongueurType avec la valeur CHAR ou BYTE.

Gérer les options du SGBD

À l'aide de la dialogue « Gérer les options du SGBD », vous pouvez rapidement modifier le Type et/ou le propriétaire du SGBD d'un object de base de données individuel ou de plusieurs objets au sein d'un Paquetage individuel ou d'une hiérarchie Paquetage . Vous pouvez également créer des index Foreign Key en masse sur toutes Foreign Keys qui n'ont pas encore d'index.

Accéder

Ruban	Conception > Paquetage > Gérer > Options du SGBD Développer > Modélisation de données > Générateur de base de données > Cliquez-droit sur la base de données souhaitée | Charger | cliquez-droit sur le nœud racine | Gérer les options du SGBD

Possibilités

Option	Action
Paquetage	Affiche le nom du Paquetage dans la fenêtre Navigateur sur lequel vous travaillez actuellement. Si nécessaire, cliquez sur le bouton et sélectionnez un autre Paquetage à l'aide de la fenêtre du navigateur (une version de la dialogue « Rechercher Paquetage »).
Inclure des objets dans Paquetages enfants	Cochez cette case pour inclure tous les objets de base de données dans tous les sous-packages. La sélection ou la désélection de ce contrôle actualisera immédiatement la liste des objets.
Liste des objets	Ce contrôle de liste affichera les noms de tous les objets du Paquetage actuel (ou de la hiérarchie Paquetage) ainsi que son alloué et son propriétaire. Par défaut, chaque object a sa case à cocher sélectionnée chaque fois que la liste est chargée ou actualisée.
Tous	Cliquez sur ce bouton pour cocher toutes les cases décochées dans la « Liste des objets ».
Aucun	Cliquez sur ce bouton pour décocher toutes les cases sélectionnées dans la « Liste des objets ».
Changer de SGBD	Cochez cette case si vous souhaitez modifier le SGBD attribué aux objets dans le Paquetage . Fournissez des valeurs pour les champs « SGBD actuel » et « Nouveau SGBD » afin de continuer. La liste déroulante « SGBD actuel » comprend l'option « <Tout> », qui modifie plusieurs valeurs de SGBD différentes vers la nouvelle valeur . Note : Lors de l'exécution de cette fonction, les types de données de toutes les colonnes Tableau sont automatiquement convertis vers la correspondance la plus proche pour le SGBD sélectionné ; par conséquent, vous devez effectuer une révision manuelle des types de données après avoir exécuté le processus.

Changer de propriétaire	Cochez cette case si vous souhaitez modifier le propriétaire des objets sélectionnés dans la « Liste des objets ». Spécifiez le propriétaire actuel dans le champ « Propriétaire actuel » afin de continuer. Laisser le champ « Nouveau propriétaire » vide supprimera la propriété Propriétaire de tous les objets sélectionnés.
Créer des index sur Foreign Keys	Cochez cette case pour créer un index sur toutes Foreign Keys du Paquetage , là où il n'en existe pas déjà.
OK	Cliquez sur ce bouton pour démarrer le processus de mise à jour. Le bouton est désactivé sauf si au moins un object de la liste et l'une des options de mise à jour sont sélectionnés.

Objets de base de données

Bien que Tableaux soient les composants fondamentaux d'une base de données relationnelle et permettent la définition de colonnes, de types de données, de clés et d'index, il existe un certain nombre d'autres objets importants dans les systèmes RDBM, notamment :

- Vues - une Vue représente l'ensemble de résultats d'une requête prédéfinie ; ils sont dérivés dynamiquement des données stockées dans un ou plusieurs Tableaux (ou autres Vues)
- Procédures - une fonctionnalité que certains produits SGBD implémentent pour fournir des sous-programmes pouvant contenir une ou plusieurs instructions SQL pour effectuer une tâche spécifique telle que la validation des données, le contrôle d'accès ou pour réduire le trafic réseau entre les clients et les serveurs SGBD.
- Fonctions - une fonctionnalité que certains produits SGBD implémentent pour fournir un mécanisme permettant d'étendre les fonctionnalités du serveur de base de données ; chacun est une routine qui peut accepter des paramètres, effectuer une action (comme un calcul complexe) et renvoyer le résultat de cette action sous forme de valeur
- Séquences - une fonctionnalité que certains produits SGBD implémentent pour fournir un mécanisme permettant de générer des valeurs uniques - la Séquence garantit que chaque appel renvoie une valeur unique

L' UML lui-même ne précise pas comment modélisation des données est effectuée, mais Enterprise Architect dispose d'un profil UML entièrement intégré pour modélisation des données et d'une gamme de fonctionnalités intégrées au produit principal qui faciliteront modélisation des données.

Le profil utilise des stéréotypes et Valeur Étiquetés pour étendre les éléments UML standard dans des constructions modélisation de données. Ceci est réalisé en ajoutant le stéréotype object de base de données à une classe UML ; pour que vous modélisiez :

- diagrammes Modélisation de données sous forme de diagrammes de classes UML étendus
- Tableaux en tant qu'objets de classe UML avec un stéréotype de « table »
- Vues en tant qu'objets de classe UML avec un stéréotype de <<view>>
- Procédures en tant qu'objets de classe UML avec un stéréotype de « procédure »
- Fonctionne comme des objets de classe UML avec un stéréotype de « fonction »
- Séquences en tant qu'objets de classe UML avec un stéréotype de « dbsequence »

Vous pouvez rapidement créer et configurer tous ces objets dans votre modèle de base de données avec Enterprise Architect .

Tableaux de base de données

Tableaux sont les composants fondamentaux d'une base de données relationnelle, représentant plusieurs lignes d'éléments de données structurés (appelés colonnes). Chaque élément de données individuel saisi dans une base de données relationnelle est représenté par une valeur dans une colonne.

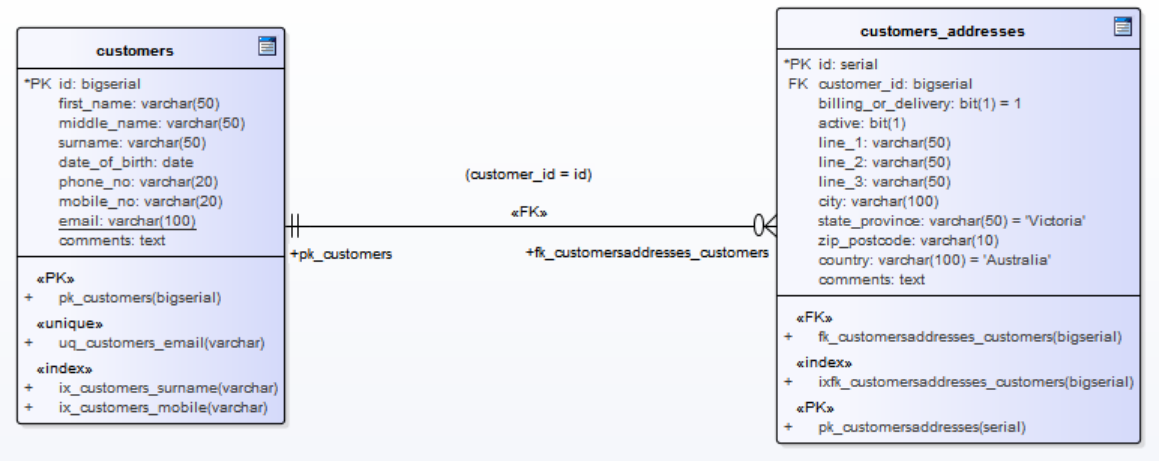
Le profil UML d' Enterprise Architect pour Modélisation des données représente :

- Tableaux de base de données en tant qu'objets de classe UML avec un stéréotype de « table »
- Colonnes Tableau en tant qu'attributs UML d'un Tableau , avec un stéréotype de « colonne »
- Primary Keys en tant qu'opérations/méthodes UML d'un Tableau , avec un stéréotype de « PK »
- Foreign Keys en tant qu'opérations/méthodes UML d'un Tableau , avec un stéréotype de <<FK>>
- Index en tant qu'opérations/méthodes UML d'un Tableau , avec un stéréotype de « index »
- Contraintes uniques comme opérations/méthodes UML d'un Tableau , avec un stéréotype de « unique »
- Contraintes de vérification en tant qu'opérations/méthodes UML d'un Tableau , avec un stéréotype de <<check>>
- Tableau Déclencheurs en tant qu'opérations/méthodes UML d'un Tableau , avec un stéréotype de « déclencheur »

Enterprise Architect fait référence à toutes les opérations UML d'un Tableau collectivement sous le nom de Contraintes. Par conséquent, l'écran que vous utilisez pour gérer les attributs et les opérations UML d'un Tableau est appelé l'écran Colonnes et contraintes.

Exemple

Cet exemple simple de Modèle de données physiques *dans* Enterprise Architect se compose de deux Tableaux de base de données représentés par des classes UML , nommés diagramme et *customer_addresses* .



Chaque Tableau définit les colonnes de la base de données, à l'aide d'attributs UML saisis de manière appropriée pour le SGBD cible (dans ce cas, PostgreSQL).

Notes

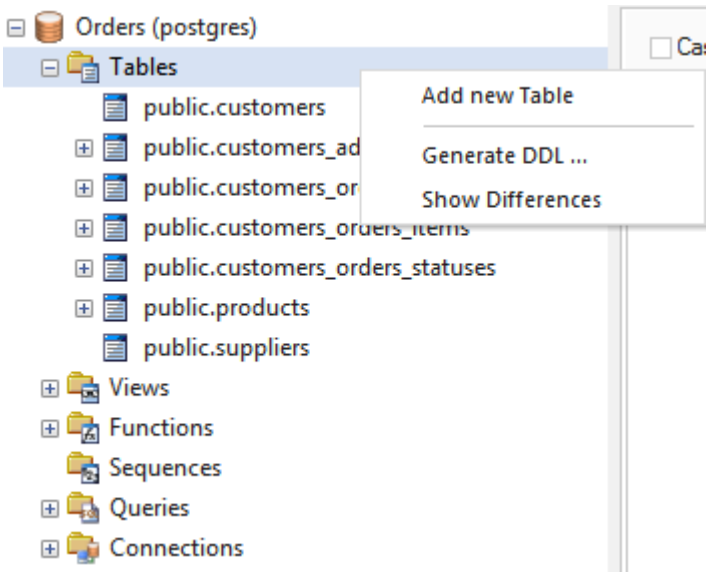
- Le stéréotype Tableau est indiqué par l'icône dans le coin supérieur droit de chaque classe (voir la rubrique *Notation Modélisation des données*)
- L'écran de maintenance Enterprise Architect pour la gestion des colonnes Tableau ne vous permet pas de modifier le stéréotype des attributs, puisque « colonne » est la seule option valide.
- Il est possible de masquer l'étiquette du stéréotype « colonne » présentée dans l'exemple Tableaux (voir le sujet *Notation Modélisation de données*)

Créer un Tableau de base de données

La création de Tableaux de base de données au sein du modèle est fondamentale pour modélisation des données. Il existe trois manières de créer un Tableau :

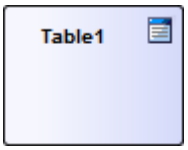
- Dans le générateur de base de données
- Sur un diagramme Data Modèle ouvert
- Utilisation de l'option Navigateur Nouvel Élément

Ajouter un Tableau de base de données avec le générateur de base de données

Étape	Action
1	Ouvrez le Database Builder ('Développement > Modélisation de données > Database Builder').
2	Chargez ou créez un modèle de données.
3	Cliquez-droit sur le Paquetage Tableaux et sélectionnez 'Ajouter un nouveau Tableau '. 
4	Remplacez le nom par défaut par le nom approprié pour Tableau et appuyez sur la touche Entrée.
5	Double-cliquez sur l'élément Tableau pour définir les propriétés Tableau .

Ajouter un Tableau de base de données à un diagramme

Étape	Action
1	Créer et/ou ouvrir un diagramme Modélisation de données.
2	Faites glisser et déposez l'icône de la boîte à outils « Tableau » sur le diagramme .

	 Cela génère un nouvel élément Tableau : 
3	Double-cliquez sur l'élément Tableau pour définir les propriétés Tableau .

Colonnes Tableau de base de données

Dans une base de données relationnelle, une colonne Tableau (parfois appelée champ) stocke une valeur de données unique d'un type particulier dans chaque ligne du Tableau . Les colonnes Tableau peuvent avoir diverses propriétés individuelles telles qu'une valeur par défaut ou si le champ accepte les valeurs Null.

Une colonne Tableau de base de données est représentée dans le profil Modélisation de données UML comme un attribut stéréotypé ; c'est-à-dire un attribut avec le stéréotype « colonne » . Dans Enterprise Architect vous définissez et gérez les colonnes Tableau à l'aide de la page « Colonnes » spécialement conçue du générateur de base de données ou de la dialogue « Colonnes et contraintes ».

Créer des colonnes Tableau de base de données

Une colonne de base de données Tableau est représentée dans le profil Modélisation de données UML comme un attribut avec le stéréotype « colonne ». Pour un Tableau sélectionné, vous pouvez révision les colonnes existantes et créer de nouvelles colonnes, sur la page « Colonnes » du générateur de base de données ou sur l'écran « Colonnes et contraintes ».

Vous pouvez définir les détails des colonnes directement sur la liste des colonnes dans l'onglet 'Colonnes'. Les modifications sont automatiquement enregistrées au fur et à mesure que vous remplissez chaque champ. Certains champs comportent certaines restrictions quant aux données que vous pouvez saisir, comme décrit ici. L'onglet contient également un panneau ' Propriétés ' et un champ ' Notes ', qui sont renseignés avec les informations existantes sur la colonne sélectionnée. Chaque nouvelle colonne que vous créez se voit automatiquement attribuer un ensemble de valeurs par défaut et est ajoutée au bas de la liste.

Accéder

Ruban	Développer > Modélisation de données > Générateur de base de données > Cliquez sur Tableau > Colonnes > Clic droit > Ajouter une nouvelle colonne
Menu Contexte	Dans diagramme , cliquez-droit sur le Tableau | Fonctionnalités | Colonnes | Clic droit | Ajouter une nouvelle colonne
Raccourcis Clavier	Sélectionnez un tableau | F9 | Touche de tabulation (pour définir le focus de saisie sur l'onglet « Colonnes ») | Ctrl+N

Créer des colonnes dans un Tableau

Option	Action
Nom	Remplacez le nom par défaut par le texte du nom de colonne approprié.
Type	Cliquez sur la flèche déroulante et sélectionnez le type de données approprié pour la colonne. Les types de données disponibles dépendent du SGBD attribué au Tableau parent.
Longueur	(Facultatif) Certains types de données ont un composant de longueur. Par exemple, VARCHAR a une longueur qui définit le nombre de caractères pouvant être stockés. Si le type de données n'a pas de composant de longueur, ce champ est désactivé. Si le champ est disponible et si vous devez définir un nombre de caractères, saisissez ici la valeur .
Échelle	(Facultatif) Certains types de données ont un composant d'échelle. Par exemple, DECIMAL a une échelle qui définit le nombre de décimales pouvant être conservées. Si le type de données n'a pas de composant d'échelle, ce champ est désactivé. Si le champ est disponible et si vous devez définir une échelle, saisissez la valeur ici.

PK	Cochez la case si la colonne fait partie de la Primary Key de ce Tableau .
Pas nul	Cochez la case si les valeurs vides sont interdites pour cette colonne. La case à cocher est désactivée si la case « PK » est cochée.
Alias	Si cela est nécessaire à des fins d'affichage et de documentation, saisissez un autre nom pour le champ.
Valeur initiale	Si nécessaire, saisissez une valeur qui peut être utilisée comme valeur par défaut pour cette colonne.
Notes	Type toute information supplémentaire nécessaire pour documenter la colonne. Vous pouvez formater le texte à l'aide de la barre d'outils Notes en haut du champ.

Propriétés des colonnes

Les propriétés appropriées pour le système de gestion de base de données de Tableau s'affichent automatiquement dans le panneau « Propriétés » (développez la branche « Colonne (<nom>) » si elles ne sont pas visibles).

Propriété	SGBD
Autonum (Incrément de Startnum)	Oracle MySQL serveur SQL DB2 PostgreSQL Notes : Si vous avez besoin d'une séquence de numérotation automatique, définissez cette propriété sur True et, si nécessaire, définissez le numéro de départ et l'incrément.
Généré	DB2 Notes : Définissez cette propriété supplémentaire pour la numérotation automatique dans DB2 sur « Par défaut » ou « Toujours ».
PasPourRep	Serveur SQL Notes : Définissez cette propriété sur True si vous souhaitez bloquer la réplication.
Remplissage zéro	MySQL Notes : Définissez cette propriété sur True ou False pour indiquer si les champs sont remplis à zéro ou non.
Non signé	MySQL Notes : Définissez cette propriété sur True ou False pour indiquer si les champs acceptent ou non les nombres non signés.
Type de longueur	Oracle Notes : Définissez cette propriété pour définir la sémantique des caractères comme 'Aucun', 'Byte' ou 'Char'.

Supprimer les colonnes Tableau de la base de données

Pour une base de données Tableau sélectionnée, vous pouvez révision les colonnes existantes et supprimer n'importe quelle colonne individuelle, dans l'onglet « Colonnes » de l'écran Colonnes et contraintes.

Accéder

Utilisez l'une des méthodes décrites ici pour afficher une liste de colonnes pour un tableau , puis sélectionnez une colonne et supprimez-la.

Lorsque vous sélectionnez l'option 'Supprimer la colonne '<nom>", si toutes les règles de validation sont satisfaites, la colonne est immédiatement supprimée.

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur Tableau > Colonnes > Cliquez-droit sur le nom de la colonne > Supprimer la colonne <nom>
Menu Contexte	Dans diagramme , cliquez-droit sur le Tableau | Fonctionnalités | Colonnes | Cliquez-droit sur le nom de la colonne | Supprimer la colonne <nom>
Raccourcis Clavier	F9 | Utilisez « Flèche vers le haut » ou « Flèche vers le bas » pour sélectionner une colonne | Ctrl+D

Notes

- Si la colonne Tableau de base de données supprimée est impliquée dans des contraintes, elle en sera automatiquement supprimée

Réorganiser les colonnes Tableau de la base de données

Si vous avez plusieurs colonnes définies dans une base de données Tableau , vous pouvez modifier l'ordre dans lequel elles sont répertoriées. L'ordre dans la liste est l'ordre dans lequel les colonnes apparaissent dans le DDL généré.

Accéder

Utilisez l'une des méthodes décrites ici pour afficher une liste de colonnes pour un Tableau , puis sélectionnez une colonne et repositionnez-la dans la liste.

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur Tableau
Menu Contexte	Dans diagramme , cliquez-droit sur le Tableau | Fonctionnalités | Colonnes
Raccourcis Clavier	F9

Changer l'ordre des colonnes

Étape	Action
1	Dans l'onglet « Colonnes », cliquez sur le nom de la colonne souhaitée dans la liste.
2	Cliquez-droit et sélectionnez le : • Option « Déplacer la colonne <nom> vers le haut » (ou appuyez sur Ctrl+Flèche vers le haut) pour déplacer la colonne d'une position vers le haut. • Option « Déplacer la colonne <nom> vers le bas » (ou appuyez sur Ctrl+Flèche vers le bas) pour déplacer la colonne d'une position vers le bas. Ces options ont un effet immédiat aussi bien dans l'onglet 'Colonnes' que sur un diagramme .

Utilisation Propriétés Tableau de base de données

Une fois que vous avez créé une base de données Tableau , vous pouvez révision ses propriétés et vérifier que les valeurs du SGBD et du propriétaire sont correctes. Pour afficher la dialogue ' Propriétés ' d'un Tableau , double-cliquez sur le nom Tableau dans le Paquetage ' Tableaux du générateur de base de données' ou sur l'élément Tableau dans un diagramme .

Important

Un SGBD doit être attribué à un Tableau avant de pouvoir y ajouter des colonnes. Si vous utilisez le Database Builder alors le SGBD du modèle de données sera automatiquement appliqué à tous les nouveaux Tableaux ; cependant, si vous avez ajouté un Tableau par d'autres moyens (par exemple en travaillant sur un diagramme), il s'agit d'une étape manuelle.

Tâches

Une fois les propriétés de la base de données Tableau définies, vous êtes prêt à ajouter des colonnes.

Tâche
Définissez le type de base de données pour un Tableau - outre le nom Tableau , la propriété la plus importante à définir pour un Tableau de base de données est le type de base de données.
Définir le propriétaire de la base de données Tableau - Pour certains SGBD, tous Tableaux doivent se voir attribuer un propriétaire/schéma ; dans Enterprise Architect cette propriété est définie comme une Valeur Étiquetée avec le nom Propriétaire.
Définir des options étendues : certains SGBD ont des options étendues qui ne concernent que ce SGBD. Ces propriétés étendues sont stockées sous Valeur Étiquetés .

SGBD par défaut

Avant de créer un Modèle de données physiques, il est conseillé de définir le SGBD par défaut, qui sera automatiquement appliqué aux nouveaux objets de base de données que vous créez en dehors du générateur de base de données. Vous pouvez définir le type de SGBD par défaut de l'une des manières suivantes :

- Sélectionnez ' Démarrer > Apparence > Préférences > Préférences > Source Code Engineering > Éditeurs de Code ', puis paramétrez le champ 'Base de données par défaut'.
- Sélectionnez « Paramètres > Données de référence > Paramètres > Types de données de base de données », puis sélectionnez un nom de produit et cochez la case « Définir par défaut ».
- Définissez le SGBD dans le deuxième champ de la barre d'outils de génération de code

Définir le Type de base de données

La propriété la plus importante à définir pour une base de données Tableau (après son nom) est le type de base de données ou SGBD. La valeur du SGBD sélectionnée contrôlera la manière dont Enterprise Architect déterminera :

- Comment le nom Tableau sera affiché (avec ou sans propriétaire)
- Quel ensemble de règles de validation sera appliqué lors modélisation de la base de données
- Les types de données disponibles lors de la création de colonnes,
- Quel ensemble de gabarits DDL sera utilisé dans la génération DDL

Accéder

Sélectionnez un Tableau dans la fenêtre Navigateur ou sur un diagramme puis, en utilisant l'une des méthodes décrites ici, ouvrez la boîte dialogue ' Propriétés ' du Tableau , affichez l'onglet 'Général', puis affichez l'onglet enfant 'Principal'.

Ruban	Conception > Élément > Editeurs > Dialogue Propriétés > Général > Principal
Menu Contexte	Cliquez-droit sur l'élément Tableau | Propriétés | Action spéciale | Général | Principal
Raccourcis Clavier	Maj+Entrée | Général | Principal
Autre	Double-cliquez sur l'élément Tableau |Général | Principal

Possibilités

Champ/Bouton	Action
Base de données	Cliquez sur le bouton déroulant et sélectionnez le type de base de données requis dans la liste.
Appliquer	Cliquez sur le bouton Appliquer pour enregistrer les modifications en attente.
OK	Cliquez sur le bouton OK pour enregistrer les modifications en attente et fermer l'écran.

Définir le propriétaire/schéma de la base de données Tableau

Pour certains SGBD, tous Tableaux doivent se voir attribuer un propriétaire/schéma. Dans Enterprise Architect cette propriété est physiquement définie comme une Valeur Étiquetée avec le nom Propriétaire. Cependant, une page de propriétés spéciale est fournie pour vous aider à gérer facilement la propriété Propriétaire.

Accéder

Sélectionnez un Tableau dans la fenêtre Navigateur ou sur un diagramme puis, en utilisant l'une des méthodes décrites ici, ouvrez la boîte dialogue « Propriétés » de Tableau , affichez l'onglet « Général » et affichez l'onglet enfant « Détail du Tableau ».

Ruban	Conception > Élément > Editeurs > Propriétés > << tableau >>
Menu Contexte	Cliquez-droit sur l'élément Tableau | Propriétés | Action spéciale > Général > Détails Tableau
Raccourcis Clavier	Maj+Entrée | Général | Détail Tableau
Autre	Double-cliquez sur l'élément Tableau |'Général' | 'Détail Tableau '

Définir le propriétaire de la base de données Tableau

Étape	Action
1	Dans le champ « Propriétaire », saisissez le nom du propriétaire ou du schéma du Tableau .

Définir les options MySQL

Pour utiliser Foreign Keys dans MySQL, vous devez déclarer le type de base de données Tableau comme InnoDB.

Déclarez le type Tableau comme InnoDB

Étape	Action
1	Ajoutez une Valeur Étiquetée nommée *Type* au Tableau .
2	Définissez le champ « Valeur » sur « InnoDB ».

Générer DDL

Lorsque vous générez du DDL pour ce Tableau , le type Tableau est inclus dans le script SQL.

Pour autoriser les versions ultérieures de MySQL, les options Tableau supplémentaires qui peuvent être ajoutées de la même manière incluent :

Étiqueter	Valeur (exemple)
MOTEUR	InnoDB
JEU DE CARACTÈRES	latin1
CHARSET	latin1
COLLATIONNER	latin1_german2_ci

Définir Propriétés Tableau de base de données Oracle

Pour définir des propriétés Oracle Database Tableau supplémentaires, vous utilisez les Valeur Étiquetés de Tableau .

Définir Propriétés

Les mêmes propriétés peuvent être ajoutées aux index et aux contraintes, en mettant en surbrillance l'index ou la contrainte Operation et en ajoutant la Valeur Étiquetés appropriée.

Étape	Action
1	Ajoutez une ou plusieurs Valeur Étiquetés au Tableau , en utilisant les noms fournis dans la colonne 'Propriété/Tag' du Tableau ' Propriétés '.
2	Spécifiez la valeur appropriée pour chaque balise. Des exemples sont fournis dans la colonne « Valeur » de ce Tableau Propriétés . • CACHE - NOCACHE • VERSION DB-9.0.111 • LISTES GRATUITES - 1 • PROPRIÉTAIRE DE LA SUBVENTION1 - SÉLECTIONNER • GRANT OWNER2 - SUPPRIMER, INSÉRER, SÉLECTIONNER, METTRE À JOUR • INITIALE - 65536 • INITRANS-1 • LOGEMENT - LOGEMENT • MAXEXTENSIONS - 2147483645 • MAXTRANS-255 • MINEXTENTS - 1 • SUIVI - SUIVI • PROPRIÉTAIRE - PROPRIÉTAIRE1 • PARALLÈLE - NON PARALLÈLE • PCTFREE - 10 • PCTINAUGMENTATION - 0 • PCTUTILISÉ - 0 • SYNONYMES - PUBLIC:TABLE_PUB;OWNER2:TABLE_OWNER2 • TABLESPACE - MON_TABLESPACE • TEMPORAIRE - OUI

Contraintes/index de la base de données Tableau

Dans Enterprise Architect , les contraintes et les index Tableau sont modélisés sur le même écran ; collectivité, on les appelle Contraintes. Les contraintes de base de données définissent les conditions imposées sur le comportement d'une base de données Tableau . Ils comprennent:

- Primary Key - identifie de manière unique un enregistrement dans un Tableau , composé d'une ou plusieurs colonnes
- Index - améliore les performances des opérations de récupération et de tri sur les données Tableau
- Contraintes uniques : une combinaison de valeurs qui identifient de manière unique une ligne dans Tableau
- Foreign Key - une colonne (ou un ensemble de colonnes) qui impose une relation entre deux Tableaux
- Vérifier les contraintes : applique l'intégrité du domaine en limitant les valeurs acceptées par une colonne.
- Tableau Déclencheur - SQL ou code exécuté automatiquement suite à la modification des données d'un Tableau

Dans Enterprise Architect , vous pouvez définir et gérer les contraintes Tableau à l'aide de la page « Contraintes/Index » spécialement conçue à cet effet du générateur de base de données ou de l'écran Colonnes et contraintes.

Accéder

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur le nom Tableau > Contraintes/Index | Cliquez-droit | Ajouter une nouvelle contrainte
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Index | Cliquez-droit | Ajouter une nouvelle contrainte
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index : Ctrl+N

Créer une contrainte

Le processus de création de l’un de ces types de contraintes est le même et est réalisé de l’une des manières décrites ici.

Créer une contrainte - À l'aide du menu contextuel ou du clavier

Étape	Action
1	Une nouvelle contrainte est automatiquement créée et se voit attribuer le nom par défaut *de contrainte n* (où *n* est un compteur) et un « Type » d'« index ». Remplacez le nom par défaut par votre propre nom de contrainte.
2	Si nécessaire, dans le champ ' Type ' cliquez sur la flèche déroulante et sélectionnez le type de contrainte approprié.
3	Si vous préférez, saisissez un alias pour la contrainte, dans le champ 'Alias'. Le champ « Colonnes » est en lecture seule ; il est rempli avec les colonnes que vous attribuez à l'onglet « Colonnes impliquées ».

Créer une contrainte - Remplacer le texte gabarit

Étape	Action
1	Dans l'onglet 'Contraintes/Index' du Tableau sélectionné, la liste des contraintes se termine par le texte gabarit *Nouvelle contrainte* . Remplacez ce texte par le nom de contrainte approprié et appuyez sur la touche Entrée.
2	La nouvelle contrainte est automatiquement créée et le Type d'index par défaut lui est attribué. Si nécessaire, dans le champ ' Type ' cliquez sur la flèche déroulante et sélectionnez le type de contrainte approprié.
3	Si vous préférez, saisissez un alias pour la contrainte, dans le champ 'Alias'. Le champ « Colonnes » est en lecture seule ; il est rempli avec les colonnes que vous attribuez à l'onglet « Colonnes impliquées ».

Attribuer des colonnes à une contrainte

Les types de contraintes Primary Key , Foreign Key , Index et Unique doivent tous avoir au moins une colonne qui leur est attribuée ; cela définit les colonnes impliquées dans la contrainte.

Étape	Action
1	Dans l'onglet « Contraintes/Index » du Tableau sélectionné, cliquez sur la contrainte à laquelle vous attribuez des colonnes.
2	Le panneau « Colonnes disponibles » répertorie toutes les colonnes définies pour Tableau . Pour chaque colonne à affecter à la contrainte, cliquez-droit sur le nom de la colonne et sélectionnez 'Attribuer la colonne <nom>'. Le nom de la colonne est transféré dans la liste « Colonnes attribuées ».

Annuler l'attribution de colonnes à une contrainte

Étape	Action
1	Dans l'onglet « Contraintes/Index » du Tableau sélectionné, cliquez sur la contrainte à partir de laquelle vous désattribuez des colonnes.
2	Dans la liste 'Colonnes affectées', cliquez-droit sur le nom de la colonne à désaffecter de la contrainte et sélectionnez 'Désaffecter la colonne <nom>'. Le nom de la colonne est transféré dans la liste « Colonnes disponibles ».

Réorganiser les colonnes affectées dans une contrainte

Si la contrainte comporte un certain nombre de colonnes, vous pouvez réorganiser la séquence en déplaçant le nom d'une colonne sélectionnée d'une place vers le haut ou vers le bas de la liste à la fois. Pour faire ça:

- Cliquez-droit sur le nom de la colonne pour vous déplacer et sélectionner soit :
  - Déplacer la colonne '<nom>' vers le haut (Ctrl+Flèche vers le haut) ou
  - Déplacer la colonne '<nom>' vers le bas (Ctrl+Flèche vers le bas)

Supprimer une contrainte

Pour supprimer une contrainte dont vous n'avez plus besoin, cliquez-droit sur le nom de la contrainte dans la liste de l'onglet 'Contraintes/Index' et sélectionnez l'option 'Supprimer la contrainte <nom>'. Si toutes les règles de validation pour le type de contrainte donné sont respectées, la contrainte est immédiatement supprimée du référentiel ainsi que toutes les relations associées (le cas échéant).

Primary Keys

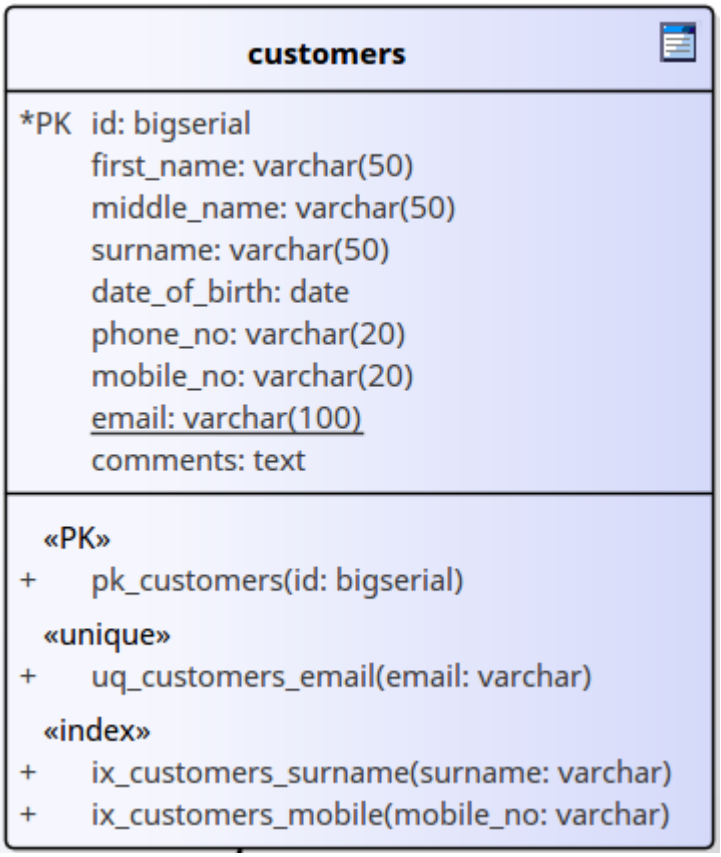
Une Primary Key est une colonne (ou un ensemble de colonnes) qui identifie de manière unique chaque enregistrement dans un Tableau . Un Tableau ne peut avoir qu'une seule Primary Key . Certains SGBD support des propriétés supplémentaires de Primary Keys , telles que Clustered ou Fill Factor.

Accéder

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur le nom Tableau
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Index

Créer une Primary Key

Dans Enterprise Architect vous pouvez créer une Primary Key à partir de l'onglet « Colonnes » ou de l'onglet « Contraintes/Index ». Dans les deux cas, lorsque vous ajoutez une colonne à une contrainte Primary Key , la colonne est automatiquement définie sur « Non nulle ». De plus, tout diagramme (en supposant que l'option « Afficher les qualificateurs et les indicateurs de visibilité » soit définie) contenant l'élément Tableau affichera le préfixe « PK » à côté du nom de la colonne. Dans cette image, voir la première colonne « id : bigserial ».



Créer une Primary Key - depuis l'onglet Colonnes

Étape	Action
1	Soit: • Dans le Database Builder, cliquez sur un Tableau avec une ou plusieurs colonnes définies, puis cliquez sur l'onglet « Colonnes », ou • Sur un diagramme , cliquez sur un Tableau et appuyez sur F9 pour afficher l'onglet 'Colonnes'
2	Pour chaque colonne à inclure dans la Primary Key , cochez la case « PK ». Si une contrainte Primary Key n'est pas préalablement définie pour le Tableau actuel, le système créera une nouvelle contrainte en utilisant le gabarit de nom Primary Key .

Créer une Primary Key - depuis l'onglet Contraintes

Étape	Action
1	Soit: • Dans le Database Builder, cliquez sur un Tableau avec une ou plusieurs colonnes définies, puis cliquez sur l'onglet « Contraintes/Index », ou • Sur un diagramme , cliquez sur un Tableau et appuyez sur F10 pour afficher l'onglet 'Contraintes/Index'
2	Remplacez le texte *de la nouvelle contrainte* par le nom Primary Key , appuyez sur la touche Entrée et cliquez sur la flèche déroulante du champ « Type », puis sélectionnez « PK ».
3	Attribuez les colonnes requises à la contrainte PK.
4	Définissez les propriétés étendues de la Primary Key à l'aide du panneau de propriétés. • Le facteur de remplissage est une valeur numérique comprise entre 0 et 100 • Is Clustered est une valeur booléenne qui détermine l'ordre physique de la façon dont les données sont stockées ; pour la plupart des SGBD, la propriété Is Clustered est définie par défaut sur True pour Primary Keys

Supprimer des colonnes d'une Primary Key

Vous pouvez supprimer des colonnes d'une Primary Key en utilisant soit l'onglet « Colonnes », soit l'onglet « Contraintes/Index ».

Supprimer des colonnes d'une Primary Key - à l'aide de l'onglet Colonnes

Étape	Action

1	Soit: • Dans le générateur de base de données, cliquez sur le Tableau avec la Primary Key , puis cliquez sur l'onglet « Colonnes », ou • Sur un diagramme , cliquez sur un Tableau et appuyez sur F9 pour afficher l'onglet 'Colonnes'
2	Pour chaque colonne que vous souhaitez supprimer de la Primary Key , décochez la case « PK ». Si vous avez supprimé toutes les colonnes de la contrainte Primary Key et que la Primary Key n'est plus nécessaire, elle doit être supprimée manuellement.

Supprimer des colonnes d'une Primary Key - à l'aide de l'onglet Contraintes/Index

Étape	Action
1	Soit: • Dans le générateur de base de données, cliquez sur le Tableau avec la Primary Key , puis cliquez sur l'onglet « Contraintes/Index », ou • Sur un diagramme , cliquez sur un Tableau et appuyez sur F10 pour afficher l'onglet 'Contraintes/Index'
2	Annulez l'affectation des colonnes sur la contrainte PK, si nécessaire.

Notes

- Avertissement : Enterprise Architect suppose que les contraintes Primary Key se voient attribuer au moins une colonne ; cependant, Enterprise Architect n'applique pas cette règle lors de modélisation
  Si le DDL est généré pour un Tableau dont Primary Key n'a aucune colonne attribuée, ce DDL ne sera pas valide.

Primary Keys non clusterisées

Lorsque vous créez une Primary Key dans certains SGBD (tels que SQL Server ou ASA), elle est automatiquement créée avec la propriété « Is Clustered » définie sur True. Par conséquent, lorsque vous modélisez une Primary Key dans un modèle de données Enterprise Architect , le même comportement se produit.

Les index clusterisés offrent des performances améliorées pour accéder aux colonnes impliquées, en organisant physiquement les données par ces colonnes. Il ne peut y avoir qu'un seul index clusterisé par Tableau .

Dans certaines situations, vous pourriez être plus intéressé par les performances des colonnes autres que celles affectées à la Primary Key et vous devrez donc modifier l'affectation par défaut afin que la Primary Key ne soit pas clusterisée.

Accéder

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur le nom Tableau > Contraintes/Index
Menu Contexte	Dans diagramme ou dans la fenêtre Navigateur | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Index
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index

Définir Primary Key comme non clusterisée

Par la suite, vous pouvez modéliser un index pour le même Tableau en cluster.

Étape	Action
1	Mettez en surbrillance la contrainte Primary Key existante. Les propriétés Primary Key s'affichent dans le panneau « Propriété ».
2	Pour la propriété *Is Clustered* , dans le champ « Valeur », cliquez sur la flèche déroulante et modifiez la valeur sur False.

Index de base de données

Les index de base de données sont appliqués aux Tableaux pour améliorer les performances des opérations de récupération et de tri des données. Plusieurs index peuvent être définis sur un Tableau ; cependant, chaque index impose des frais généraux (sous forme de temps de traitement et de stockage) au serveur de base de données pour les maintenir à mesure que des informations sont ajoutées et supprimées de Tableau .

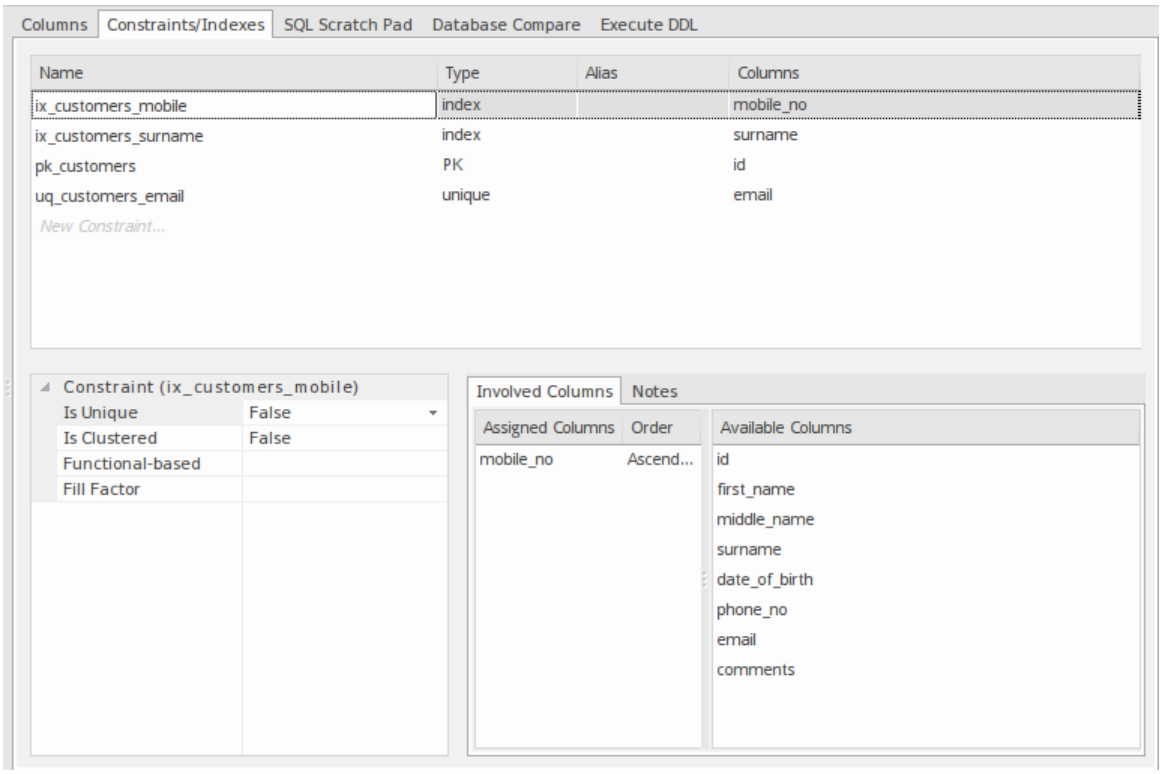
Dans Enterprise Architect un index est modélisé comme une opération stéréotypée.

Certains SGBD support des types d'index spéciaux ; Enterprise Architect les définit à l'aide de propriétés supplémentaires telles que basées sur les fonctions, clusterisées et facteurs de remplissage.

Accéder

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur le nom Tableau > Contraintes/Index
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Index
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index

Travailler sur un index



Étape	Action
1	Dans l'onglet 'Contraintes/Index' du Tableau , cliquez-droit et sélectionnez 'Ajouter une nouvelle

	contrainte'. La nouvelle contrainte est ajoutée avec le nom par défaut « contrainte1 » et le Type « index ». Remplacez le nom par votre nom d'index préféré.
2	Attribuez les colonnes appropriées à l'index. La liste « Colonnes affectées » comporte un champ « Ordre » supplémentaire qui spécifie l'ordre (croissant ou décroissant) dans lequel chaque colonne attribuée est stockée dans l'index. Vous pouvez modifier l'ordre de chaque colonne, selon vos besoins. De plus, pour les index MySQL, un champ « Len » sera visible dans lequel vous pourrez définir des index partiels ; c'est-à-dire un index qui utilise le premier nombre de caractères « n » d'un champ texte. Le champ « Len » ne prend que des valeurs numériques entières comprises entre 0 et la longueur définie de la colonne. Une valeur de 0 (qui est la valeur par défaut) indique que la colonne entière doit être indexée.
3	Dans le panneau 'Propriété', révision les paramètres des propriétés étendues définies pour le SGBD actuel.

Propriétés supplémentaires

Propriété	Description
Est unique	(Vrai / Faux) indique si l'index actuel est un 'Index Unique'. Un index unique garantit que la ou les colonnes indexées ne contiennent pas de valeurs en double, garantissant ainsi que chaque ligne a une valeur unique (ou une combinaison de valeurs lorsque l'index est constitué de plusieurs colonnes).
Est groupé	(Vrai / Faux) indique si l'index actuel est un « index clusterisé ». Avec un index clusterisé, les lignes du tableau sont physiquement stockées dans le même ordre que dans l'index, il ne peut donc y avoir qu'un seul index clusterisé par tableau . Par défaut, la Primary Key d'un tableau est clusterisée. Tous les SGBD ne prennent pas support les index clusterisés. Par conséquent, la propriété d'index « Est clusterisé » ne sera visible que pour les SGBD qui la support .
Est-ce qu'un bitmap	(Vrai / Faux) indique si l'index actuel est un index 'Bitmap'. Les index bitmap sont destinés à être utilisés sur des colonnes qui ont relativement peu de valeurs uniques (appelées colonnes à « faible cardinalité ») et qui sont physiquement constituées d'un tableau de bits (communément appelé bitmaps) pour chaque valeur unique. Chacun des tableaux aura un bit pour chaque ligne du tableau . Prenons cet exemple : un index bitmap est créé sur une colonne appelée « Gender », qui comporte les options « Mâle » ou « Femme ». Physiquement, l'index sera composé de deux tableaux de bits, un pour « Mâle » et un pour « Femme ». Le tableau de bits femelle aura un 1 dans chaque bit où la ligne correspondante a la valeur « Femme ». Les propriétés « Is Bitmap » et « Is Unique » s'excluent mutuellement et la génération DDL ignorera donc la propriété « Is Unique » lorsque la propriété « Is Bitmap » est True. Les index bitmap ne sont pris en charge que par Oracle ; par conséquent, cette propriété n'est visible que lors modélisation des index Oracle.
Facteur de remplissage	Une valeur numérique comprise entre 0 et 100, qui définit le pourcentage d'espace disponible qui doit être utilisé pour les données.

	Tous les SGBD ne support le facteur de remplissage, par conséquent la propriété d'index « Fill Factor » ne sera visible que pour les SGBD qui la support .
Basé sur le fonctionnel	Une instruction SQL qui définit la fonction/instruction qui sera évaluée et les résultats indexés ; Par exemple: INFÉRIEUR("champ") Tous les SGBD ne prennent pas support les index fonctionnels. Par conséquent, la propriété Index « Functional-based » ne sera visible que pour les SGBD qui les support , tels que PostgreSQL et Oracle.
Inclure	Identifie une liste CSV de colonnes non clés du tableau actuel. Tous les SGBD ne prennent pas support la propriété « Inclure » sur les index. Cette propriété ne sera donc visible que pour les SGBD qui la support .

Notes

- Avertissement : Enterprise Architect suppose qu'au moins une colonne est attribuée aux index ; cependant, Enterprise Architect n'applique pas cette règle lors de modélisation
  Si le DDL est généré pour un Tableau dont l'index est défini sans colonne(s) attribuée(s), ce DDL ne sera pas valide, sauf si l'index est basé sur les fonctionnalités.
- Toutes les colonnes affectées à un index fonctionnel sont ignorées

Contraintes uniques

Les contraintes uniques imposent le « caractère unique » d'un ensemble de champs dans toutes les lignes d'un Tableau , ce qui signifie que deux lignes d'un Tableau ne peuvent pas avoir les mêmes valeurs dans les champs d'une contrainte unique. Les contraintes uniques sont similaires aux Primary Keys (dans le sens où elles imposent également « l'unicité »), mais la principale différence est qu'un Tableau peut avoir plusieurs contraintes uniques définies mais une seule Primary Key .

Accéder

Ruban	Développer > Modélisation des données > Générateur de base de données > Cliquez sur le nom Tableau > Contraintes/Index > Cliquez-droit > Ajouter une nouvelle contrainte
Menu Contexte	Dans diagramme ou dans la fenêtre Navigateur | Cliquez-droit sur l'élément Tableau | Fonctionnalités | Contraintes/Index
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index : Ctrl+N

Créer une contrainte

Étape	Action
1	Dans l'onglet 'Contraintes/Index', une nouvelle contrainte est automatiquement créée et se voit attribuer le nom de contrainte par défaut et un ' Type ' d'index. Remplacez le nom de la contrainte par un nom qui l'identifie comme une contrainte unique.
2	Dans le champ ' Type ', changez la valeur de 'index' à 'unique'.

Notes

- Avertissement : Enterprise Architect suppose qu'au moins une colonne est affectée aux contraintes uniques ; cependant, Enterprise Architect n'applique pas cette règle lors de modélisation
  Si le DDL est généré pour un Tableau qui a une contrainte unique définie sans colonne(s) attribuée(s), ce DDL ne sera pas valide.

Foreign Keys

Une Foreign Key définit une colonne (ou une collection de colonnes) qui applique une relation entre deux Tableaux . Il est de la responsabilité du serveur de base de données de faire respecter cette relation afin de garantir l'intégrité des données. La définition de modèle d'une Foreign Key consiste en un Tableau parent (primaire) contenant un ensemble unique de données qui est ensuite référencé dans un Tableau enfant (étranger).

Dans Enterprise Architect , une Foreign Key est modélisée avec deux composants UML différents (mais liés) :

- Une contrainte Foreign Key (une opération UML avec le stéréotype de « FK ») stockée sur le Tableau enfant et
- Un connecteur Association (stéréotype de « FK ») définissant la relation entre les deux Tableaux

Créer une Foreign Key

Bien que la définition d'une Foreign Key puisse être complexe, l'écran Contrainte Foreign Key simplifie la modélisation des Foreign Keys . Cet écran est spécialement conçu pour vous aider à sélectionner la contrainte du Tableau parent à utiliser et fera automatiquement correspondre les colonnes Tableau enfants à celles du Tableau parent qui font partie de la contrainte. Différents aspects du processus de développement d'une Foreign Key sont décrits ici séparément à titre d'illustration, mais le processus global doit être une transition en douceur.

Un certain nombre de conditions doivent être remplies avant qu'une définition Foreign Key puisse être enregistrée :

- Les deux Tableaux doivent avoir des SGBD correspondants définis
- Le Tableau parent doit avoir au moins une colonne
- Le Tableau parent doit avoir une Primary Key , une contrainte unique ou un index unique défini

Créer une Foreign Key - à l'aide du générateur de base de données

Étape	Action
1	Dans l'arborescence Database Builder, cliquez-droit sur le nom Tableau enfant et cliquez sur 'Ajouter une nouvelle Foreign Key sur <nom tableau >'. Une dialogue s'affiche répertoriant tous les Tableaux parents possibles.
2	Double-cliquez sur le nom Tableau parent requis dans la liste ou sélectionnez-le et cliquez sur le bouton OK . L'écran « Contrainte Foreign Key » s'affiche.

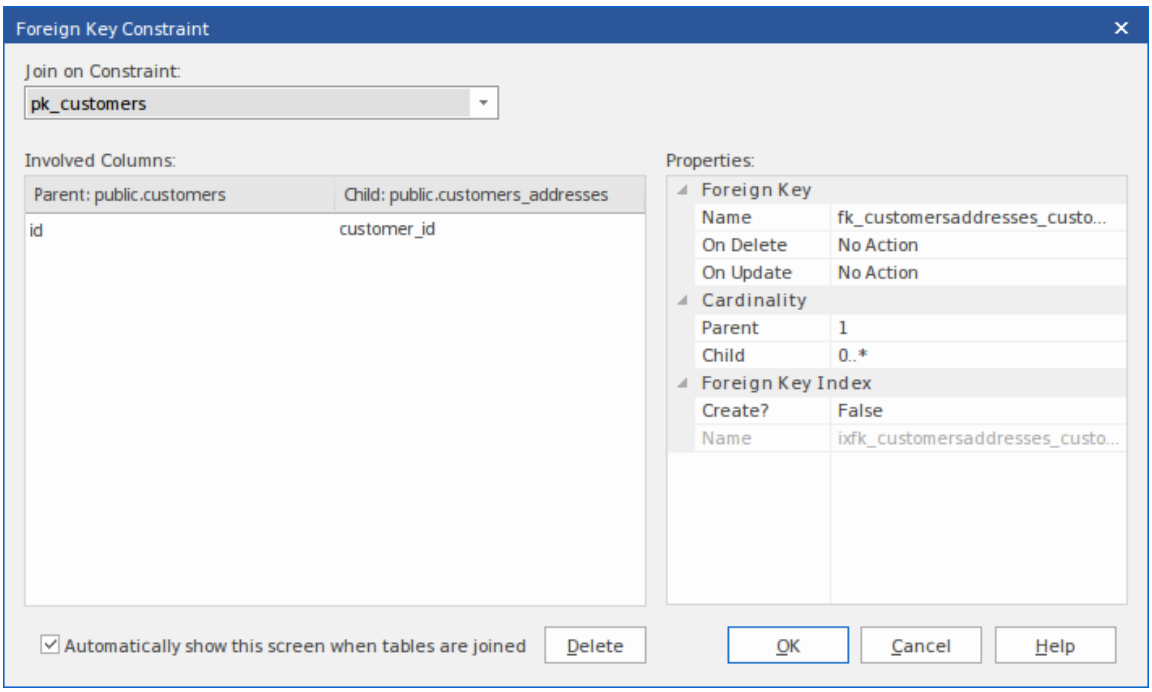
Créer une Foreign Key - en utilisant une relation sur un diagramme

Étape	Action
1	Dans le diagramme Modélisation des données, localisez le Tableau enfant (Foreign Key) et Tableau parent (Primary Key) requis.
2	Sélectionnez un connecteur Association dans la page ' Modélisation des données' de la Boîte à outils Diagramme .

3	Cliquez sur le Tableau enfant et dessinez le connecteur vers le Tableau parent.
4	Si l'écran Contrainte Foreign Key a été configuré pour s'afficher automatiquement lorsque deux Tableaux sont joints, il affiche maintenant . Sinon, soit : • Double-cliquez sur le connecteur ou • Cliquez-droit sur le connecteur et sélectionnez l'option ' Foreign Keys ' L'écran Contrainte Foreign Key s'affiche.

L'écran de contrainte Foreign Key

À titre d'exemple, cette image montre l'écran de contrainte Foreign Key chargé avec les détails de « fk_customersaddresses_customers » (tel que défini dans l'exemple de modèle).

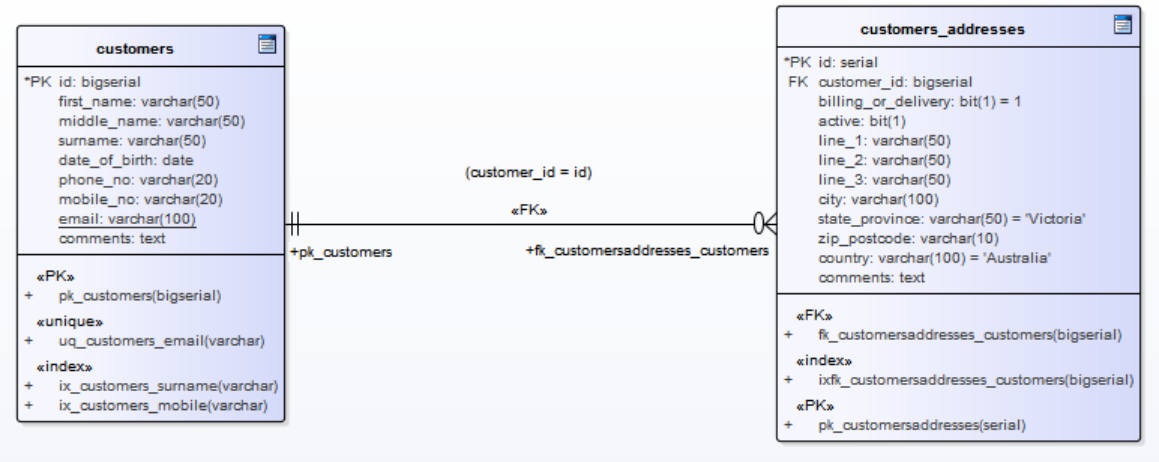


Option	Action
Rejoindre sur contrainte	Cette zone de liste déroulante répertorie toutes les contraintes définies dans le Tableau parent qui pourraient être utilisées comme base d'une Foreign Key . (Ces contraintes peuvent être Primary Keys , des contraintes uniques ou des index uniques.) La première contrainte de la liste est sélectionnée par défaut ; si ce n'est pas la contrainte souhaitée, sélectionnez la contrainte correcte dans la zone de liste déroulante. Lorsque vous sélectionnez la contrainte, ses colonnes sont automatiquement répertoriées dans le panneau « Colonnes impliquées », sous la colonne « Parent : <tablename> ».
Colonnes impliquées	Cette liste est divisée en deux : les colonnes impliquées dans la contrainte sélectionnée sont répertoriées à gauche, et les colonnes enfants qui vont être appariées aux colonnes parents sont répertoriées à droite. Lorsqu'une contrainte est sélectionnée (dans le champ 'Joindre sur contrainte') le côté parent est actualisé pour afficher toutes les colonnes affectées à la contrainte

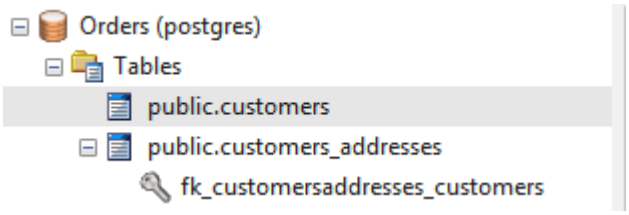
	sélectionnée. Du côté enfant, le système tentera automatiquement de faire correspondre chaque colonne parent à une colonne du même nom dans le Tableau enfant. Si le Tableau enfant n'a pas de colonne du même nom, une nouvelle colonne de ce nom sera ajoutée à la liste, marquée par (*) pour indiquer qu'une nouvelle colonne sera créée dans Tableau . Cependant, si vous souhaitez forcer le couplage avec une colonne Tableau enfant existante ou une nouvelle colonne avec un nom différent, cliquez sur le champ du nom de la colonne et soit : • Type le nom de remplacement, ou • Sélectionnez une colonne existante (cliquez sur la flèche déroulante et sélectionnez le nom dans la liste)
Nom	Ce champ définit le nom de la contrainte Foreign Key et prend par défaut un nom construit par le nom Foreign Key Gabarit . Pour changer le nom en quelque chose d'autre que le nom par défaut, écrasez simplement la valeur .
Lors de la suppression	Sélectionnez l'action qui doit être effectuée sur les données du Tableau enfant lorsque les données du Tableau parent sont supprimées, afin de maintenir l'intégrité référentielle.
En mise à jour	Sélectionnez l'action à entreprendre sur les données du Tableau enfant lorsque les données du Tableau parent sont mises à jour, afin de maintenir l'intégrité référentielle.
Parent	Cliquez sur la flèche déroulante et sélectionnez la cardinalité du Tableau parent dans la Foreign Key .
Enfant	Cliquez sur la flèche déroulante et sélectionnez la cardinalité du Tableau enfant dans la Foreign Key .
Créer?	Si vous souhaitez créer un index Foreign Key en même temps que la Foreign Key , définissez cette propriété sur True. Le nom de l'index Foreign Key est contrôlé par le gabarit de l'index Foreign Key et le nom généré est affiché dans le champ « Nom » sous le champ « Créer ? champ.
Afficher automatiquement cet écran lorsque tableaux sont joints	(Pour modélisation schématique) Cochez cette case pour afficher automatiquement cet écran à chaque fois qu'une Association est créée entre deux Tableaux .
Supprimer	Cliquez sur ce bouton pour supprimer la Foreign Key existante (enregistrée) actuellement sélectionnée. Une prompt s'affiche pour confirmer la suppression (et la suppression de l'index Foreign Key , s'il existe) - cliquez sur le bouton Oui. La suppression d'une Foreign Key laisse en place un connecteur d'association, que vous pouvez modifier ou supprimer (cliquez-droit et sélectionnez 'Supprimer l'association : à < Nom Tableau >').
OK	Cliquez sur ce bouton pour enregistrer la Foreign Key .

Exemples

Cet exemple montre Foreign Keys simples dans un diagramme :



La même Foreign Key sera affichée dans l'arborescence du générateur de base de données en tant que nœud enfant sous « customer.addresses » de Tableau .



Vérifier les contraintes

Une contrainte de vérification renforce l'intégrité du domaine en limitant les valeurs acceptées par une colonne.

Accéder

Ruban	Développer > Modélisation des données > Générateur de base de données > Cliquez sur le nom Tableau > Contraintes/Index > Cliquez-droit > Ajouter une nouvelle contrainte
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Index | Cliquez-droit | Ajouter une nouvelle contrainte
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index : Ctrl+N

Créer une contrainte

Étape	Action
1	Dans l'onglet « Contraintes/Index » de l'écran Colonnes et contraintes, une nouvelle contrainte est automatiquement créée et se voit attribuer le nom de contrainte par défaut et un « Type » d'index. Remplacez le nom de la contrainte par un nom qui identifie la contrainte en tant que contrainte de vérification, tel que « CHK_ColumnName » (le préfixe CHK_ est facultatif).
2	Dans le champ ' Type ', changez la valeur de 'index' à 'check'.
3	Dans le panneau ' Propriétés ' de la propriété Condition, tapez l'instruction SQL qui sera utilisée comme condition de vérification ; par exemple, colonne1 < 1 000. Si la condition est longue, cliquez sur le bouton pour afficher un éditeur SQL (avec coloration syntaxique).

Supprimer une contrainte de vérification

Si vous ne souhaitez pas conserver de contrainte de contrôle, soit :

- Cliquez-droit dessus dans la liste et sélectionnez 'Supprimer la contrainte <nom>', ou
- Cliquez sur l'élément et appuyez sur Ctrl+D

La contrainte est immédiatement supprimée.

Notes

- Toutes les colonnes affectées à une contrainte de vérification sont ignorées

Tableau Déclencheurs

Un déclencheur Tableau est du SQL ou du code qui est automatiquement exécuté suite à la modification de données dans une base de données Tableau . Déclencheurs sont hautement personnalisables et peuvent être utilisés de différentes manières ; par exemple, ils pourraient être utilisés pour empêcher l'exécution de certaines activités de base de données pendant les heures de bureau, ou pour fournir une validation ou effectuer des suppressions dans les Tableaux secondaires lorsqu'un enregistrement dans le Tableau principal est supprimé.

Dans Enterprise Architect , un déclencheur Tableau est modélisé comme une opération stéréotypée et géré à l'aide de l'écran « Contraintes » de Tableau .

Accéder

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur le nom Tableau > Contraintes/Index | Cliquez-droit | Ajouter une nouvelle contrainte
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Indices | Cliquez-droit | Ajouter une nouvelle contrainte
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index : Ctrl+N

Créer un Tableau Déclencheur

Étape	Action
1	Dans l'onglet 'Contraintes/Index', une nouvelle contrainte est automatiquement créée et se voit attribuer le nom de contrainte par défaut et un ' Type ' d'index. Remplacez le nom de la contrainte par un nom qui identifie la contrainte comme un déclencheur , tel que TRG_OnCustomerUpdate. (Le préfixe TRG_ est facultatif.)
2	Dans le champ ' Type ', changez la valeur de 'index' à ' déclencheur '.
3	Dans le panneau ' Propriétés ' de la propriété Statement, saisissez l'instruction SQL complète (y compris CREATE TRIGGER) qui définira le Déclencheur . Si la condition est longue, cliquez sur le bouton pour afficher un éditeur SQL (avec coloration syntaxique).
4	Les propriétés Déclencheur Time et Déclencheur Event sont actuellement des valeurs à titre informatif et ne sont pas utilisées dans la génération DDL.

Supprimer un Tableau Déclencheur

Si vous ne souhaitez pas conserver de déclencheur , soit :

- Cliquez-droit dessus dans la liste et sélectionnez 'Supprimer la contrainte <nom>', ou

- Cliquez sur l'élément et appuyez sur Ctrl+D

Le déclencheur est immédiatement supprimé.

Notes

- Toutes les colonnes affectées au tableau déclencheurs sont ignorées

Vues de base de données

Une Database Vue représente les résultats d'une requête prédéfinie. Contrairement à un Tableau , une Vue est dérivée dynamiquement des données d'un ou plusieurs Tableaux (ou autres Vues). Enterprise Architect supporte la définition de Vues avec et sans cette instruction :

Instruction "Créer Vue {viewName} As"

Le système l'ajoutera automatiquement dynamiquement (s'il est manquant) chaque fois que la génération DDL est effectuée. L'avantage de ne pas définir cette instruction est que lorsqu'un object vue est renommé, la propriété « Vue Definition » n'a pas besoin d'être mise à jour manuellement.

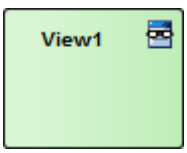
Vous pouvez créer une Database Vue soit :

- Dans le générateur de base de données ou
- En faisant glisser l'icône ' Vue ' de la Data Modélisation Toolbox sur un diagramme

Ajouter une Vue de base de données avec le générateur de base de données

Étape	Action
1	Ouvrez le générateur de base de données.
2	Chargez ou créez un modèle de données.
3	Cliquez-droit sur le Paquetage ' Vues ' et sélectionnez 'Ajouter une nouvelle Vue '.
4	Remplacez le nom par défaut par le nom approprié pour Vue et appuyez sur la touche Entrée.
5	Double-cliquez sur la nouvelle Vue , ou cliquez-droit dessus et sélectionnez 'SQL Object Propriétés '. La dialogue « Éditeur Object SQL » s'affiche.

Ajouter une Vue de base de données à un diagramme

Étape	Action
1	Ouvrez votre diagramme Modélisation Données et, si nécessaire, affichez la page ' Modélisation de Données' de la Boîte à Diagramme (cliquez sur pour afficher la dialogue ' Trouvez Item de Boîte à Outils ' et précisez ' Modélisation de Données').
2	Faites glisser l'icône ' Vue ' sur le diagramme . Cela génère l'élément Vue : 

3	Cliquez-droit sur le nouvel élément Vue et sélectionnez 'SQL Object Propriétés '. La dialogue « Éditeur Object SQL » s'affiche.

Éditeur Object SQL

La dialogue « Éditeur Object SQL » est partagée par un certain nombre d'objets de base de données SQL (Vues , Procédures, Fonctions et Séquences) ; il aide le modélisateur de données à gérer les différentes propriétés de l' object basé sur SQL.

Option	Action
Base de données	S'il a déjà été défini, le type de base de données par défaut s'affiche. Si la valeur par défaut n'a pas été définie ou si vous souhaitez modifier le type de base de données pour cette Vue , cliquez sur la flèche déroulante et sélectionnez le SGBD cible à modéliser.
Dépendances	Une liste d'objets dont dépend l' object actuel. La liste « Dépendances » affiche : • Chaque connecteur Dépend entre cette Vue et un autre Tableau ou Vue • Tous les noms object (spécifiés sous forme de liste CSV) dans les Valeur Étiquetés 'parents'
Notes	Si nécessaire, saisissez un commentaire sur la Vue actuelle.
Définition	Type la définition complète de SQL Vue . Pour les versions d' Enterprise Architect jusqu'à 12.1 (Build 1227), cela doit inclure la syntaxe CREATE_VIEW appropriée pour le SGBD cible (pour les versions ultérieures, cela n'est pas nécessaire). Par exemple: CRÉER UNE VUE 'MyViewName' AS [voir la définition] L'éditeur de code fournit Intelli-sense pour les mots-clés SQL de base, les fonctions et les noms de tous les objets du modèle de données actuel.

Procédures de base de données

Les procédures de base de données (parfois appelées procédures stockées ou Procs) sont des sous-programmes qui peuvent contenir une ou plusieurs instructions SQL qui effectuent une tâche spécifique. Ils peuvent être utilisés pour la validation des données, le contrôle d'accès ou pour réduire le trafic réseau entre les clients et les serveurs SGBD. Une logique métier étendue et complexe peut être intégrée au sous-programme, offrant ainsi de meilleures performances.

Les procédures de base de données sont similaires aux fonctions de base de données. La principale différence réside dans la manière dont elles sont invoquées : les fonctions de base de données peuvent être utilisées de la même manière que pour toute autre expression dans les instructions SQL, tandis que les procédures de base de données doivent être invoquées à l'aide de l'instruction CALL ou EXEC, selon le SGBD.

Dans Enterprise Architect , les procédures de base de données peuvent être modélisées de deux manières :

- En tant qu'objets individuels (la méthode par défaut) ou
- Comme opérations dans un conteneur

Fonctionnellement, les deux méthodes aboutissent à la production du même DDL. La principale différence est visuelle : en ayant plusieurs opérations dans un seul conteneur, vous avez moins d'éléments et moins d'encombrement sur le diagramme .

Objets individuels

Les procédures de base de données modélisées sous forme d'objets individuels sont des classes UML avec le stéréotype « procédure » ; vous les créez soit :

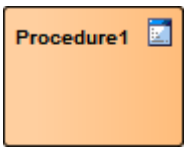
- Dans le générateur de base de données ou
- En faisant glisser l'icône 'Procédure' de la Boîte à Outils Modélisation de Données sur un diagramme

Ajouter une procédure de base de données à l'aide du générateur de base de données

Étape	Action
1	Ouvrez le générateur de base de données.
2	Chargez ou créez un modèle de données.
3	Cliquez-droit sur le Paquetage de Procédures et sélectionnez 'Ajouter une Nouvelle Procédure'.
4	Remplacez le nom par défaut par le nom approprié pour la procédure et appuyez sur la touche Entrée.
5	Double-cliquez sur la nouvelle procédure, ou cliquez-droit dessus et sélectionnez 'SQL Object Propriétés '. L'écran de l'éditeur Object SQL s'affiche.

Ajouter une procédure de base de données à un diagramme

Étape	Action

1	Ouvrez votre diagramme Modélisation Données et, si nécessaire, affichez la page ' Modélisation de Données' de la Boîte à Diagramme (cliquez sur pour afficher la dialogue ' Trouvez Item de Boîte à Outils ' et précisez ' Modélisation de Données').
2	Faites glisser l'icône 'Procédure' sur le diagramme . Cela génère l'élément Procedure : 
3	Cliquez-droit sur le nouvel élément Procedure et sélectionnez 'SQL Object Propriétés '. L'écran de l'éditeur Object SQL s'affiche.

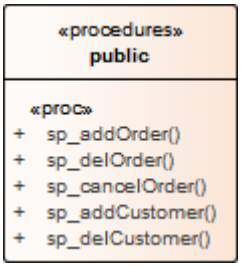
Éditeur Object SQL

La dialogue « Éditeur Object SQL » est partagée par un certain nombre d'objets de base de données basés sur SQL (Vues , Procédures et Fonctions) ; il vous aide à gérer les différentes propriétés de l' object basé sur SQL.

Option	Action
Base de données	S'il a déjà été défini, le type de base de données par défaut s'affiche. Si la valeur par défaut n'a pas été définie ou si vous souhaitez modifier le type de base de données pour cette procédure, cliquez sur la flèche déroulante et sélectionnez le SGBD cible à modéliser.
Notes	Si nécessaire, saisissez un commentaire sur la procédure en cours.
Définition	Type la définition complète de la procédure SQL, y compris la syntaxe CREATE PROCEDURE. L'éditeur de code fournit Intelli-sense pour les mots-clés SQL de base, les fonctions et les noms de tous les objets du modèle de données actuel.

Opérations dans un conteneur

Les procédures de base de données modélisées comme des opérations ont un object conteneur, celui-ci étant une classe UML avec le stéréotype « procédures » (avec un « s » à la fin). Chaque procédure de base de données est une opération de stéréotype « proc ». Le système fournit une fenêtre de maintenance dédiée à travers laquelle vous pouvez facilement gérer les procédures de base de données définies comme opérations.


«procedures»
public
«proc»
+ sp_addOrder()
+ sp_delOrder()
+ sp_cancelOrder()
+ sp_addCustomer()
+ sp_delCustomer()

Fonctions de base de données

Les fonctions de base de données vous fournissent un mécanisme pour étendre les fonctionnalités du serveur de base de données. Une fonction de base de données est une routine qui accepte des paramètres, effectue une action (telle qu'un calcul complexe) et renvoie le résultat de cette action sous forme de valeur . Selon la fonction, la valeur de retour peut être soit une valeur unique, soit un ensemble de résultats.

Une fois créée, une fonction de base de données peut être utilisée comme expression dans une instruction SQL.

Dans Enterprise Architect , les fonctions de base de données peuvent être modélisées de deux manières :

- En tant qu'objets individuels (la méthode par défaut) ou
- Comme opérations dans un conteneur

Fonctionnellement, les deux méthodes aboutissent à la production du même DDL. La principale différence est visuelle : en ayant plusieurs opérations dans un seul conteneur, vous avez moins d'éléments et moins d'encombrement sur le diagramme .

Objets individuels

Les fonctions de base de données modélisées sous forme d'objets individuels sont des classes UML avec le stéréotype « fonction » ; vous les créez soit :

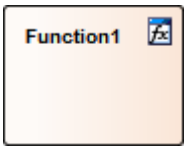
- Dans le générateur de base de données ou
- En faisant glisser l'icône Fonction de la Boîte à Outils Modélisation de Données sur un diagramme

Ajouter une fonction de base de données à l'aide du générateur de base de données

Étape	Action
1	Ouvrez le générateur de base de données.
2	Chargez ou créez un modèle de données.
3	Cliquez-droit sur le Paquetage de fonctions et sélectionnez 'Ajouter une nouvelle fonction'.
4	Remplacez le nom par défaut par le nom approprié pour la fonction et appuyez sur la touche Entrée.
5	Double-cliquez sur la nouvelle fonction, ou cliquez-droit dessus et sélectionnez 'SQL Object Propriétés '. L'écran de l'éditeur Object SQL s'affiche.

Ajouter une fonction de base de données à un diagramme

Étape	Action
1	Ouvrez votre diagramme Modélisation Données et, si nécessaire, affichez la page ' Modélisation de Données' de la Boîte à Diagramme (cliquez sur pour afficher la dialogue ' Trouvez Item de Boîte à

	Outils ' et précisez ' Modélisation de Données').
2	Faites glisser l'icône 'Fonction' sur le diagramme . Cela génère l'élément Function : 
3	Cliquez-droit sur le nouvel élément Function et sélectionnez 'SQL Object Propriétés '. L'écran de l'éditeur Object SQL s'affiche.

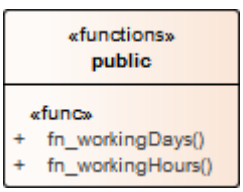
Éditeur Object SQL

La dialogue « Éditeur Object SQL » est partagée par un certain nombre d'objets de base de données basés sur SQL (Vues , Procédures et Fonctions) ; il vous aide à gérer les différentes propriétés de l' object basé sur SQL.

Option	Action
Base de données	S'il a déjà été défini, le type de base de données par défaut s'affiche. Si la valeur par défaut n'a pas été définie ou si vous souhaitez modifier le type de base de données pour cette fonction, cliquez sur la flèche déroulante et sélectionnez le SGBD cible à modéliser.
Notes	Si nécessaire, saisissez un commentaire sur la fonction actuelle.
Définition	Type la définition complète de la fonction SQL, y compris la syntaxe CREATE FUNCTION. L'éditeur de code fournit Intelli-sense pour les mots-clés SQL de base, les fonctions et les noms de tous les objets du modèle de données actuel.

Opérations dans un conteneur

Les fonctions de base de données modélisées comme des opérations ont un object conteneur, celui-ci étant une classe UML avec le stéréotype « fonctions » (avec un « s » à la fin). Chaque Fonction est une opération de stéréotype « func ». Le système fournit une fenêtre de maintenance dédiée à travers laquelle vous pouvez facilement gérer les fonctions de base de données stockées en tant qu'opérations.



Séquences de base de données

Les séquences sont une fonctionnalité que certains produits SGBD implémentent pour fournir aux utilisateurs un mécanisme permettant de générer des valeurs uniques - la Séquence garantit que chaque appel renvoie une valeur unique. Ceci est particulièrement important lorsque le résultat de la Séquence est utilisé comme Primary Key . Ceux-ci peuvent être générés avec un schéma à charger sur le serveur SGBD.

Les séquences sont fournies afin que les utilisateurs de la base de données ne soient pas obligés d'implémenter leur propre générateur valeur unique. Tous les produits SGBD ne prennent pas support les séquences ; ceux qui ne fournissent pas de fonctionnalité permettant aux colonnes d'être initialisées avec une valeur incrémentielle.

Dans Enterprise Architect , les séquences peuvent être modélisées de deux manières :

- En tant qu'objets individuels (la méthode par défaut) ou
- Comme opérations dans un conteneur

Fonctionnellement, les deux méthodes aboutissent à la production du même DDL. La principale différence est visuelle : en ayant plusieurs opérations dans un seul conteneur, vous avez moins d'éléments et moins d'encombrement sur le diagramme .

Objets individuels

Les séquences modélisées comme des objets individuels sont des classes UML avec le stéréotype « dbsequence » ; vous les créez soit :

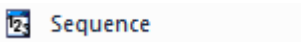
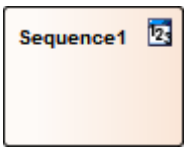
- Dans le générateur de base de données ou
- En faisant glisser l'icône ' Séquence ' des pages Toolbox ' Data Modélisation ' sur un diagramme

Ajouter une Séquence de base de données à l'aide du générateur de base de données

Étape	Action
1	Ouvrez le générateur de base de données.
2	Chargez ou créez un modèle de données.
3	Cliquez-droit sur le Paquetage Séquences et sélectionnez 'Ajouter une Nouvelle Séquence '.
4	Remplacez le nom par défaut par le nom approprié pour la Séquence et appuyez sur la touche Entrée.
5	Double-cliquez sur la nouvelle Séquence , ou cliquez-droit dessus et sélectionnez 'SQL Object Propriétés '. La dialogue « Éditeur Object SQL » s'affiche.

Ajouter une Séquence de base de données à un diagramme

Étape	Action

1	Ouvrez votre diagramme Modélisation Données et, si nécessaire, affichez la page ' Modélisation de Données' de la Boîte à Diagramme (cliquez sur pour afficher la dialogue ' Trouvez Item de Boîte à Outils ' et précisez ' Modélisation de Données').
2	Faites glisser l'icône ' Séquence ' sur le diagramme .  Cela génère l'élément Séquence : 
3	Cliquez-droit sur le nouvel élément Séquence et sélectionnez 'SQL Object Propriétés '. La dialogue « Éditeur Object SQL » s'affiche.

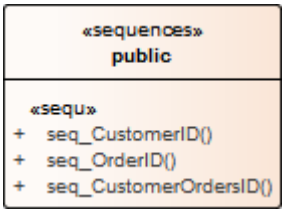
Éditeur Object SQL

La dialogue « Éditeur Object SQL » est partagée par un certain nombre d'objets de base de données basés sur SQL (Vues , Procédures et Fonctions) ; il vous aide à gérer les différentes propriétés de l' object basé sur SQL.

Option	Action
Base de données	S'il a déjà été défini, le type de base de données par défaut s'affiche. Si la valeur par défaut n'a pas été définie ou si vous souhaitez modifier le type de base de données pour cette Séquence , cliquez sur la flèche déroulante et sélectionnez le SGBD cible à modéliser.
Notes	Si nécessaire, saisissez un commentaire sur la Séquence actuelle.
Définition	Type la définition complète Séquence SQL, y compris la syntaxe CREATE SEQUENCE. L'éditeur de code fournit Intelli-sense pour les mots-clés SQL de base, les fonctions et les noms de tous les objets du modèle de données actuel.

Opérations dans un conteneur

Les séquences de base de données modélisées comme des opérations ont un object conteneur, celui-ci étant une classe UML avec le stéréotype « séquences » (avec un « s » à la fin). Chaque Séquence est une opération de stéréotype « sequ ». Le système fournit une fenêtre de maintenance dédiée à travers laquelle le modélisateur peut facilement gérer les séquences définies comme opérations.


«sequences»
public
«sequ»
+ seq_CustomerID()
+ seq_OrderID()
+ seq_CustomerOrdersID()

Requêtes SQL de base de données

Un object Query SQL fournit un mécanisme pratique pour stocker une instruction SQL dans le référentiel, en vue d'une exécution répétée sur une ou plusieurs bases de données actives.

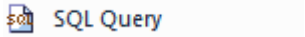
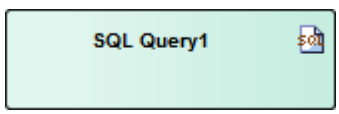
Un élément SQL Query est représenté dans le Profil Modélisation de Données UML comme un élément Artefact avec le stéréotype « sqlquery »>. Vous pouvez créer ces éléments soit :

- Dans le générateur de base de données ou
- En faisant glisser l'icône ' Query SQL' depuis les pages de la Toolbox ' Modélisation des Données' sur un diagramme

Ajouter une Query SQL de base de données à l'aide du générateur de base de données

Étape	Action
1	Ouvrez le générateur de base de données.
2	Chargez ou créez un modèle de données.
3	Cliquez-droit sur le Paquetage de requêtes et sélectionnez 'Ajouter une nouvelle Query SQL'.
4	Remplacez le nom par défaut par le nom approprié pour la Query et appuyez sur la touche Entrée.
5	Cliquez-droit sur le nouvel élément et sélectionnez 'Editer'. L'onglet « SQL Scratch Pad » s'affiche, sur lequel vous pouvez créer l'instruction Query SQL.
6	Lorsque vous avez terminé l'instruction SQL, cliquez sur le bouton Enregistrer dans Query SQL dans la barre d'outils pour enregistrer les modifications apportées à l'élément de requête.

Ajouter une fonction de base de données à un diagramme

Étape	Action
1	Ouvrez votre diagramme Modélisation Données et, si nécessaire, affichez la page ' Modélisation de Données' de la Boîte à Diagramme (cliquez sur pour afficher la dialogue ' Trouvez Item de Boîte à Outils ' et précisez ' Modélisation de Données').
2	Faites glisser l'icône « Query SQL » sur le diagramme .  Cela génère l'élément SQL Query Artifact : 
3	Double-cliquez sur le nouvel élément et mettez à jour le nom de l'élément et les autres propriétés si

	nécessaire. Pour modifier l'instruction SQL de l'élément, accédez au générateur de base de données, cliquez sur l'élément dans le Paquetage de requêtes et modifiez la Query dans l'onglet « SQL Scratch Pad ».

Créer des conteneurs d'opérations

Bien que la méthode par défaut de modélisation des fonctions, procédures et séquences de base de données consiste à les créer en tant qu'éléments individuels, vous pouvez également représenter un certain nombre de chaque type de structure en tant qu'opérations d'une classe conteneur. Vous ajoutez un stéréotype à la classe, qui spécifie :

- Le type de structure de données que la classe contiendra
- Le stéréotype qui sera automatiquement attribué à chaque opération créée dans la Classe (pour une structure de données donnée, les opérations ne peuvent être que d'un seul stéréotype)

Accéder

Boîte à outils	Faites glisser l'icône « Classe » sur le diagramme

Créer la classe conteneur

Étape	Action
1	Cliquez-droit sur l'élément Classe sur le diagramme et sélectionnez l'option 'Conception > Elément > Editeurs > Propriétés Dialogue '. La boîte dialogue ' Propriétés ' de l'élément s'affiche, affichant l'onglet 'Général'.
2	Dans le champ « Nom », saisissez un nom approprié pour le conteneur.
3	Dans le champ 'Stéréotype' (dans le tableau à droite du le dialogue) tapez : • 'fonctions' pour un conteneur de fonctions de base de données • « procédures » pour un conteneur de procédures stockées • 'séquences' pour un conteneur Séquence Le « s » à la fin du nom du stéréotype est important.
4	Cliquez sur le bouton OK pour enregistrer le paramètre et fermer le dialogue .

Créer des structures de base de données en tant qu'opérations de la classe

Étape	Action
1	Cliquez sur l'élément Classe sur le diagramme et appuyez sur F10. La boîte dialogue « Base de données <Structure> conteneur : <Classname> » s'affiche.
2	Cliquez-droit dans la liste 'Fonctions' ('Procédures' ou 'Séquences') et sélectionnez 'Ajouter une nouvelle <structure>'.

3	Dans le champ « Nom », saisissez un nom approprié pour l'opération, tel que : • fn_WorkDays • sp_AddOrder ou • seq_AddressID
4	Dans le champ « Notes » saisissez tout commentaire ou explication de l'opération. Dans le champ 'Définition de fonction' (ou 'Définition de procédure', ou 'Définition Séquence ') saisissez le texte approprié.
5	Répétez les étapes 2 à 4 jusqu'à ce que vous ayez défini toutes les opérations dont vous avez besoin.
6	Cliquez sur la liste puis sur le bouton Fermer pour fermer le dialogue et afficher les opérations au sein de la Classe sur le diagramme et dans la fenêtre Navigateur .

Paquetages Oracle

Paquetages Oracle sont des objets de base de données uniques au SGBD Oracle. Ce sont des conteneurs qui regroupent des objets logiquement liés dans une seule définition. Paquetages comportent deux parties : une spécification et un corps. Le:

- La section Spécification déclare les différents composants
- La section Corps fournit les définitions complètes des composants

Les composants Paquetage peuvent être constitués de types, de variables, de constantes, d'exceptions, de curseurs et de sous-programmes.

Dans Enterprise Architect , un Paquetage Oracle est modélisé comme une classe UML avec un stéréotype de « package » . Il comporte deux opérations :

- Spécification
- Corps

Pour chacune de ces opérations, la syntaxe SQL complète est contenue dans le champ 'Code Initial'.

Créer un Paquetage Oracle

Étape	Action
1	Ajoutez un élément Class à votre modèle de données.
2	Ouvrez la fenêtre Propriétés de l'élément et, dans le champ 'Stéréotype', saisissez la valeur ' Paquetage '.
3	Cliquez sur l'élément et appuyez sur F10, pour afficher la fenêtre Fonctionnalités sur la page 'Opérations'. Pour la spécification Paquetage , appuyez sur Ctrl+N et créez une opération portant le nom ' Spécification ' et sans type de retour.
4	La fenêtre Propriétés affiche les propriétés de l'opération ; cliquez sur l'onglet « Code » et saisissez l'intégralité des spécifications Paquetage dans le panneau de texte.
5	Revenez à la fenêtre Fonctionnalités à la page 'Opérations' et, pour le corps Paquetage , appuyez sur Ctrl+N et créez une opération avec le nom 'Corps' et sans type de retour.
6	Dans la fenêtre Propriétés , cliquez sur l'onglet « Code » et saisissez l'intégralité du code du corps Paquetage dans le panneau de texte.

Connexions à la base de données

Un object Connexion à la base de données constitue un moyen pratique de stocker les détails de connexion d'une base de données active. Enterprise Architect supporte la définition d'un certain nombre de types de connexion différents :

- Accès MS
- Oiseau de feu
- SQLite (introduit dans Enterprise Architect v16)
- Connexion native (introduite dans Enterprise Architect v16), et
- ODBC

Pour les connexions basées sur des fichiers (MS Access, Firebird et SQLite), il vous suffit de spécifier le chemin complet des fichiers de base de données. Pour les connexions natives, vous serez invité à fournir les détails de connexion à un serveur de base de données. Pour les connexions de type ODBC, vous êtes invité à effectuer une sélection dans la liste des DSN ODBC prédéfinis sur votre ordinateur.

Créer un élément de connexion à la base de données

Un élément de connexion à la base de données est représenté dans le profil Modélisation de données UML comme un élément Artifact avec le stéréotype « connexion à la base de données ». Vous les créez soit :

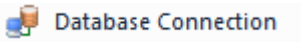
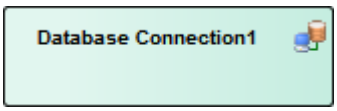
- Dans le générateur de base de données ou
- En faisant glisser l'icône 'Connexion à la base de données' depuis les pages de la boîte à outils ' Modélisation des données' sur un diagramme

Ajouter une connexion à une base de données à l'aide du générateur de base de données

Étape	Action
1	Ouvrez le générateur de base de données.
2	Chargez ou créez un modèle de données.
3	Cliquez-droit sur le Paquetage de connexions et sélectionnez « Ajouter une nouvelle connexion à la base de données ».
4	Remplacez le nom par défaut par le nom approprié pour la connexion et appuyez sur la touche Entrée.
5	Double-cliquez sur la nouvelle connexion, ou cliquez-droit dessus et sélectionnez 'DB Connection Propriétés '. La dialogue ' Propriétés de connexion à la base de données' s'affiche.

Ajouter une connexion à une base de données à un Diagramme

Étape	Action

1	Ouvrez votre diagramme Modélisation Données et, si nécessaire, affichez la page ' Modélisation de Données' de la Boîte à Diagramme (cliquez sur pour afficher la dialogue ' Trouvez Item de Boîte à Outils ' et précisez ' Modélisation de Données').
2	Faites glisser l'icône « Connexion à la base de données » sur le diagramme .  Cela génère l'élément de connexion à la base de données. 
3	Double-cliquez sur le nouvel élément. La dialogue ' Propriétés de connexion à la base de données' s'affiche.

Propriétés de connexion à la base de données

Option	Action
Type de SGBD	Cliquez sur le bouton radio correspondant au type approprié : • Base de données basée sur des fichiers MS Access • Base de données basée sur des fichiers Firebird • Base de données basée sur des fichiers SQLite • Connexion native directe, ou • Base de données basée sur ODBC Le message « Enregistrer le mot de passe ? » La case à cocher est uniquement activée pour les types de connexion ODBC et indique si Enterprise Architect doit stocker le mot de passe pour le DSN ODBC sélectionné. La case à cocher est sélectionnée par défaut ; c'est-à-dire que les mots de passe sont enregistrés. Bien que tous les mots de passe de connexion soient cryptés avant d'être enregistrés, il peut arriver que les modélisateurs de données souhaitent restreindre l'accès aux seuls utilisateurs disposant des autorisations requises.
Nom de fichier/DSN	Si vous avez sélectionné un « Type de SGBD » MS Access ou Firebird, saisissez ou recherchez l'emplacement et le nom d'un fichier physique. Si le fichier n'existe pas déjà, il sera créé. Si vous avez sélectionné un « Type de SGBD » ODBC, saisissez ou sélectionnez un DSN ODBC défini. En fonction du SGBD, vous serez peut-être invité à fournir d'autres détails tels que le serveur, ID utilisateur de connexion et le mot de passe.
Autres schémas	Ce champ agit comme un filtre de schéma pour limiter le nombre d'objets renvoyés par les requêtes effectuées sur la connexion ODBC. La saisie d'une valeur dans ce champ est particulièrement importante pour les bases de données Oracle afin de réduire le temps nécessaire à l'établissement des connexions à la base de données, en raison du grand nombre d'objets système. Si vous devez saisir plusieurs schémas à filtrer, séparez-les par des virgules.
OK	Cliquez sur ce bouton pour enregistrer les modifications que vous avez apportées.

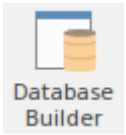
Supprimer la connexion

Si une connexion n'est plus nécessaire, vous pouvez la supprimer comme pour tout autre élément du Database Builder, de la fenêtre du Navigateur ou d'un diagramme . Cliquez-droit sur l'élément et sélectionnez l'option 'Supprimer <nom de l'élément>' correspondante.

Notes

- Il est conseillé que lorsque vous travaillez dans un environnement d'équipe (c'est-à-dire plusieurs utilisateurs partageant un seul référentiel Enterprise Architect), tous les objets de connexion à la base de données basés sur ODBC soient définis comme « sans DSN » afin que l' object de connexion à la base de données contienne tous les détails nécessaires et puisse donc être partagé entre tous les utilisateurs, bien qu'une connexion native le fasse et soit plus facile à configurer
- Le type de SGBD d'un object Connexion à la base de données ne peut pas être modifié une fois la sélection initiale enregistrée

Le constructeur de base de données

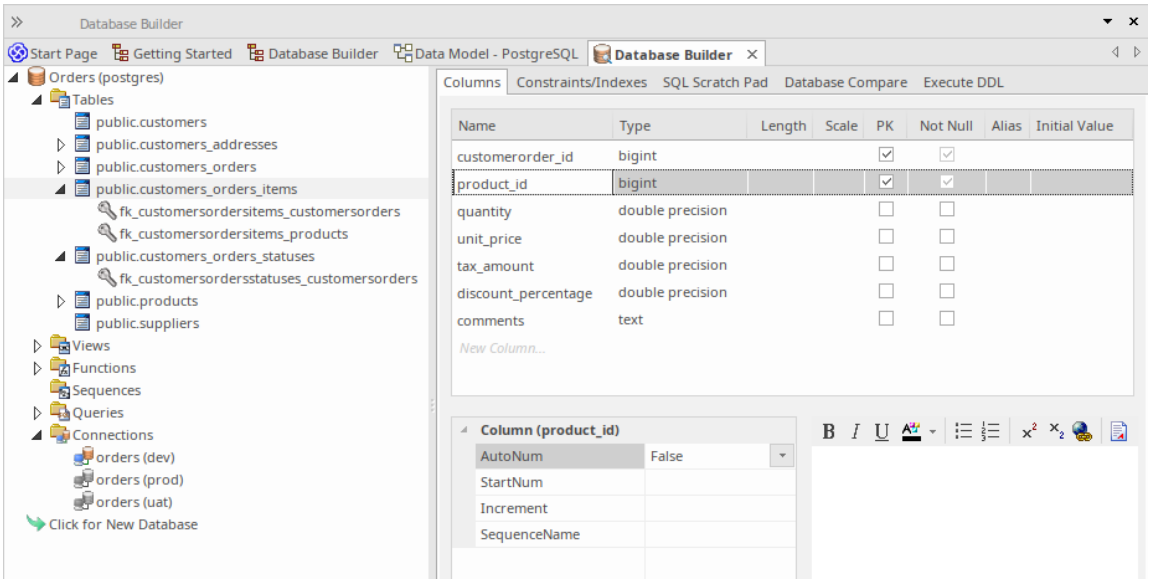


Le Database Builder est une interface sur mesure pour le modélisateur de données ; toutes les tâches modélisation liées aux bases de données peuvent être effectuées en un seul endroit. L'interface et ses écrans associés incluent uniquement les informations pertinentes pour modélisation des données, rationalisant et simplifiant ainsi le processus modélisation .

Accéder

Ruban	Développer > Modélisation de données > Générateur de base de données

Générateur de base de données



Cette figure montre le générateur de base de données chargé avec le modèle de données « Commandes (postgres) » tel qu'il apparaît dans l'exemple de modèle.

Aperçu

L'interface du Database Builder se compose de :

- Une arborescence de modèles de données, répertoriant tous les modèles de données définis dans le référentiel actuel
- Une « colonnes » onglet à travers lequel vous gérez directement les colonnes Tableau
- A 'Contraintes/Index' onglet pour la gestion directe des contraintes Tableau telles que Primary Keys , Foreign Keys et les index
- Un bloc-notes SQL que vous pouvez utiliser pour exécuter des requêtes SQL ad hoc sur une base de données active

- Un onglet « Database Compare » qui affiche les résultats des comparaisons entre le modèle de données et une base de données active
- Un onglet « Exécuter le DDL » sur lequel vous pouvez exécuter instantanément le DDL généré sur une base de données en direct

Vous pouvez utiliser le générateur de base de données pour :

- Créer, modifier et supprimer des objets de base de données (Tableaux , Vues , Procédures, Séquences et Fonctions)
- Créer, modifier et supprimer des contraintes Tableau (Primary Keys , Index, Contraintes uniques, Contraintes de vérification et Déclencheurs)
- Créer, modifier et supprimer Foreign Keys Tableau
- Informations sur le schéma de base de données d'ingénierie inverse
- Générer du DDL à partir d'une base de données modélisée
- Comparez un schéma de base de données en direct avec une base de données modélisée
- Exécuter le DDL généré sur une base de données en direct
- Exécuter des instructions SQL ad hoc sur une base de données active

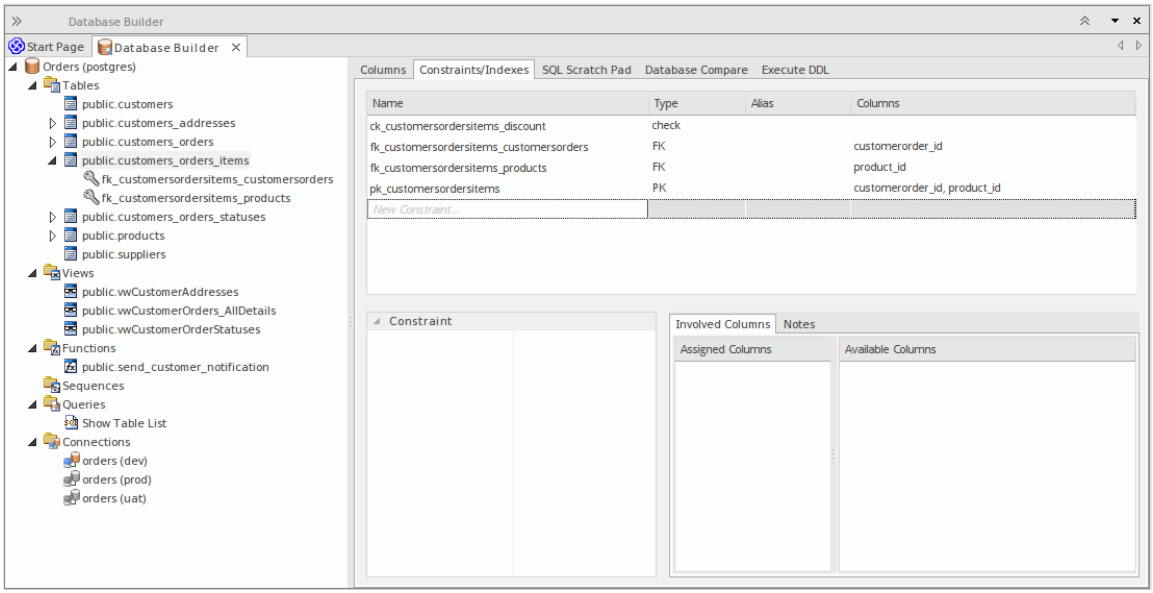
Notes

- Le générateur de base de données est disponible dans les éditions Corporate , Unified et Ultimate d' Enterprise Architect

Contraintes/Index

Tableaux sont l' object fondamental de la base de données, et les contraintes et les index (et leurs propriétés) sont la deuxième fonctionnalité Tableau la plus fréquemment modifiée, mise à jour et modifiée par les modélisateurs de données. Par conséquent, la page « Contraintes/index » est idéalement située en tant que deuxième page du générateur de base de données. interface.

Une fois qu'un Tableau est sélectionné dans l'arborescence de Database Builder, la page « Contraintes/Index » est renseignée avec la liste de contraintes et d'index actuellement définis pour le Tableau sélectionné. Le modélisateur de données peut ensuite apporter des modifications aux propriétés principales directement dans la liste. Lorsque le modélisateur de données sélectionne des contraintes ou des index individuels dans la liste, les propriétés étendues de la contrainte (et les commentaires) sont affichées immédiatement sous la liste, permettant la modification de ces propriétés étendues.



Cette figure montre l'interface Database Builder affichant l'arborescence des objets et l'onglet Colonnes affichant les colonnes du tableau sélectionné.

Notes

- La page « Contraintes/Index » ne sera remplie que lorsqu'un élément Tableau dans l'arborescence du générateur de base de données est sélectionné.

Contraintes/index de la base de données Tableau

Dans Enterprise Architect , les contraintes et les index Tableau sont modélisés sur le même écran ; collectivité, on les appelle Contraintes. Les contraintes de base de données définissent les conditions imposées sur le comportement d'une base de données Tableau . Ils comprennent:

- Primary Key - identifie de manière unique un enregistrement dans un Tableau , composé d'une ou plusieurs colonnes
- Index - améliore les performances des opérations de récupération et de tri sur les données Tableau
- Contraintes uniques : une combinaison de valeurs qui identifient de manière unique une ligne dans Tableau
- Foreign Key - une colonne (ou un ensemble de colonnes) qui impose une relation entre deux Tableaux
- Vérifier les contraintes : applique l'intégrité du domaine en limitant les valeurs acceptées par une colonne.
- Tableau Déclencheur - SQL ou code exécuté automatiquement suite à la modification des données d'un Tableau

Dans Enterprise Architect , vous pouvez définir et gérer les contraintes Tableau à l'aide de la page « Contraintes/Index » spécialement conçue à cet effet du générateur de base de données ou de l'écran Colonnes et contraintes.

Accéder

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur le nom Tableau > Contraintes/Index | Cliquez-droit | Ajouter une nouvelle contrainte
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Index | Cliquez-droit | Ajouter une nouvelle contrainte
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index : Ctrl+N

Créer une contrainte

Le processus de création de l’un de ces types de contraintes est le même et est réalisé de l’une des manières décrites ici.

Créer une contrainte - À l'aide du menu contextuel ou du clavier

Étape	Action
1	Une nouvelle contrainte est automatiquement créée et se voit attribuer le nom par défaut *de contrainte n* (où *n* est un compteur) et un « Type » d'« index ». Remplacez le nom par défaut par votre propre nom de contrainte.
2	Si nécessaire, dans le champ ' Type ' cliquez sur la flèche déroulante et sélectionnez le type de contrainte approprié.
3	Si vous préférez, saisissez un alias pour la contrainte, dans le champ 'Alias'. Le champ « Colonnes » est en lecture seule ; il est rempli avec les colonnes que vous attribuez à l'onglet « Colonnes impliquées ».

Créer une contrainte - Remplacer le texte gabarit

Étape	Action
1	Dans l'onglet 'Contraintes/Index' du Tableau sélectionné, la liste des contraintes se termine par le texte gabarit *Nouvelle contrainte* . Remplacez ce texte par le nom de contrainte approprié et appuyez sur la touche Entrée.
2	La nouvelle contrainte est automatiquement créée et le Type d'index par défaut lui est attribué. Si nécessaire, dans le champ ' Type ' cliquez sur la flèche déroulante et sélectionnez le type de contrainte approprié.
3	Si vous préférez, saisissez un alias pour la contrainte, dans le champ 'Alias'. Le champ « Colonnes » est en lecture seule ; il est rempli avec les colonnes que vous attribuez à l'onglet « Colonnes impliquées ».

Attribuer des colonnes à une contrainte

Les types de contraintes Primary Key , Foreign Key , Index et Unique doivent tous avoir au moins une colonne qui leur est attribuée ; cela définit les colonnes impliquées dans la contrainte.

Étape	Action
1	Dans l'onglet « Contraintes/Index » du Tableau sélectionné, cliquez sur la contrainte à laquelle vous attribuez des colonnes.
2	Le panneau « Colonnes disponibles » répertorie toutes les colonnes définies pour Tableau . Pour chaque colonne à affecter à la contrainte, cliquez-droit sur le nom de la colonne et sélectionnez 'Attribuer la colonne <nom>'. Le nom de la colonne est transféré dans la liste « Colonnes attribuées ».

Annuler l'attribution de colonnes à une contrainte

Étape	Action
1	Dans l'onglet « Contraintes/Index » du Tableau sélectionné, cliquez sur la contrainte à partir de laquelle vous désattribuez des colonnes.
2	Dans la liste 'Colonnes affectées', cliquez-droit sur le nom de la colonne à désaffecter de la contrainte et sélectionnez 'Désaffecter la colonne <nom>'. Le nom de la colonne est transféré dans la liste « Colonnes disponibles ».

Réorganiser les colonnes affectées dans une contrainte

Si la contrainte comporte un certain nombre de colonnes, vous pouvez réorganiser la séquence en déplaçant le nom d'une colonne sélectionnée d'une place vers le haut ou vers le bas de la liste à la fois. Pour faire ça:

- Cliquez-droit sur le nom de la colonne pour vous déplacer et sélectionner soit :
  - Déplacer la colonne '<nom>' vers le haut (Ctrl+Flèche vers le haut) ou
  - Déplacer la colonne '<nom>' vers le bas (Ctrl+Flèche vers le bas)

Supprimer une contrainte

Pour supprimer une contrainte dont vous n'avez plus besoin, cliquez-droit sur le nom de la contrainte dans la liste de l'onglet 'Contraintes/Index' et sélectionnez l'option 'Supprimer la contrainte <nom>'. Si toutes les règles de validation pour le type de contrainte donné sont respectées, la contrainte est immédiatement supprimée du référentiel ainsi que toutes les relations associées (le cas échéant).

Primary Keys

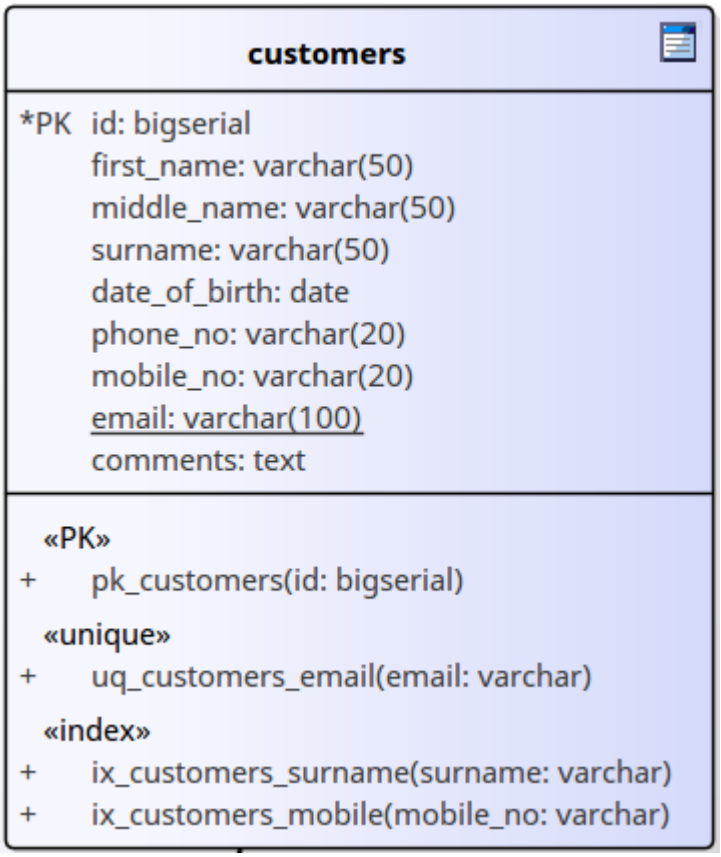
Une Primary Key est une colonne (ou un ensemble de colonnes) qui identifie de manière unique chaque enregistrement dans un Tableau . Un Tableau ne peut avoir qu'une seule Primary Key . Certains SGBD support des propriétés supplémentaires de Primary Keys , telles que Clustered ou Fill Factor.

Accéder

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur le nom Tableau
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Index

Créer une Primary Key

Dans Enterprise Architect vous pouvez créer une Primary Key à partir de l'onglet « Colonnes » ou de l'onglet « Contraintes/Index ». Dans les deux cas, lorsque vous ajoutez une colonne à une contrainte Primary Key , la colonne est automatiquement définie sur « Non nulle ». De plus, tout diagramme (en supposant que l'option « Afficher les qualificateurs et les indicateurs de visibilité » soit définie) contenant l'élément Tableau affichera le préfixe « PK » à côté du nom de la colonne. Dans cette image, voir la première colonne « id : bigserial ».



Créer une Primary Key - depuis l'onglet Colonnes

Étape	Action
1	Soit: • Dans le Database Builder, cliquez sur un Tableau avec une ou plusieurs colonnes définies, puis cliquez sur l'onglet « Colonnes », ou • Sur un diagramme , cliquez sur un Tableau et appuyez sur F9 pour afficher l'onglet 'Colonnes'
2	Pour chaque colonne à inclure dans la Primary Key , cochez la case « PK ». Si une contrainte Primary Key n'est pas préalablement définie pour le Tableau actuel, le système créera une nouvelle contrainte en utilisant le gabarit de nom Primary Key .

Créer une Primary Key - depuis l'onglet Contraintes

Étape	Action
1	Soit: • Dans le Database Builder, cliquez sur un Tableau avec une ou plusieurs colonnes définies, puis cliquez sur l'onglet « Contraintes/Index », ou • Sur un diagramme , cliquez sur un Tableau et appuyez sur F10 pour afficher l'onglet 'Contraintes/Index'
2	Remplacez le texte *de la nouvelle contrainte* par le nom Primary Key , appuyez sur la touche Entrée et cliquez sur la flèche déroulante du champ « Type », puis sélectionnez « PK ».
3	Attribuez les colonnes requises à la contrainte PK.
4	Définissez les propriétés étendues de la Primary Key à l'aide du panneau de propriétés. • Le facteur de remplissage est une valeur numérique comprise entre 0 et 100 • Is Clustered est une valeur booléenne qui détermine l'ordre physique de la façon dont les données sont stockées ; pour la plupart des SGBD, la propriété Is Clustered est définie par défaut sur True pour Primary Keys

Supprimer des colonnes d'une Primary Key

Vous pouvez supprimer des colonnes d'une Primary Key en utilisant soit l'onglet « Colonnes », soit l'onglet « Contraintes/Index ».

Supprimer des colonnes d'une Primary Key - à l'aide de l'onglet Colonnes

Étape	Action

1	Soit: • Dans le générateur de base de données, cliquez sur le Tableau avec la Primary Key , puis cliquez sur l'onglet « Colonnes », ou • Sur un diagramme , cliquez sur un Tableau et appuyez sur F9 pour afficher l'onglet 'Colonnes'
2	Pour chaque colonne que vous souhaitez supprimer de la Primary Key , décochez la case « PK ». Si vous avez supprimé toutes les colonnes de la contrainte Primary Key et que la Primary Key n'est plus nécessaire, elle doit être supprimée manuellement.

Supprimer des colonnes d'une Primary Key - à l'aide de l'onglet Contraintes/Index

Étape	Action
1	Soit: • Dans le générateur de base de données, cliquez sur le Tableau avec la Primary Key , puis cliquez sur l'onglet « Contraintes/Index », ou • Sur un diagramme , cliquez sur un Tableau et appuyez sur F10 pour afficher l'onglet 'Contraintes/Index'
2	Annulez l'affectation des colonnes sur la contrainte PK, si nécessaire.

Notes

- Avertissement : Enterprise Architect suppose que les contraintes Primary Key se voient attribuer au moins une colonne ; cependant, Enterprise Architect n'applique pas cette règle lors de modélisation
  Si le DDL est généré pour un Tableau dont Primary Key n'a aucune colonne attribuée, ce DDL ne sera pas valide.

Index de base de données

Les index de base de données sont appliqués aux Tableaux pour améliorer les performances des opérations de récupération et de tri des données. Plusieurs index peuvent être définis sur un Tableau ; cependant, chaque index impose des frais généraux (sous forme de temps de traitement et de stockage) au serveur de base de données pour les maintenir à mesure que des informations sont ajoutées et supprimées de Tableau .

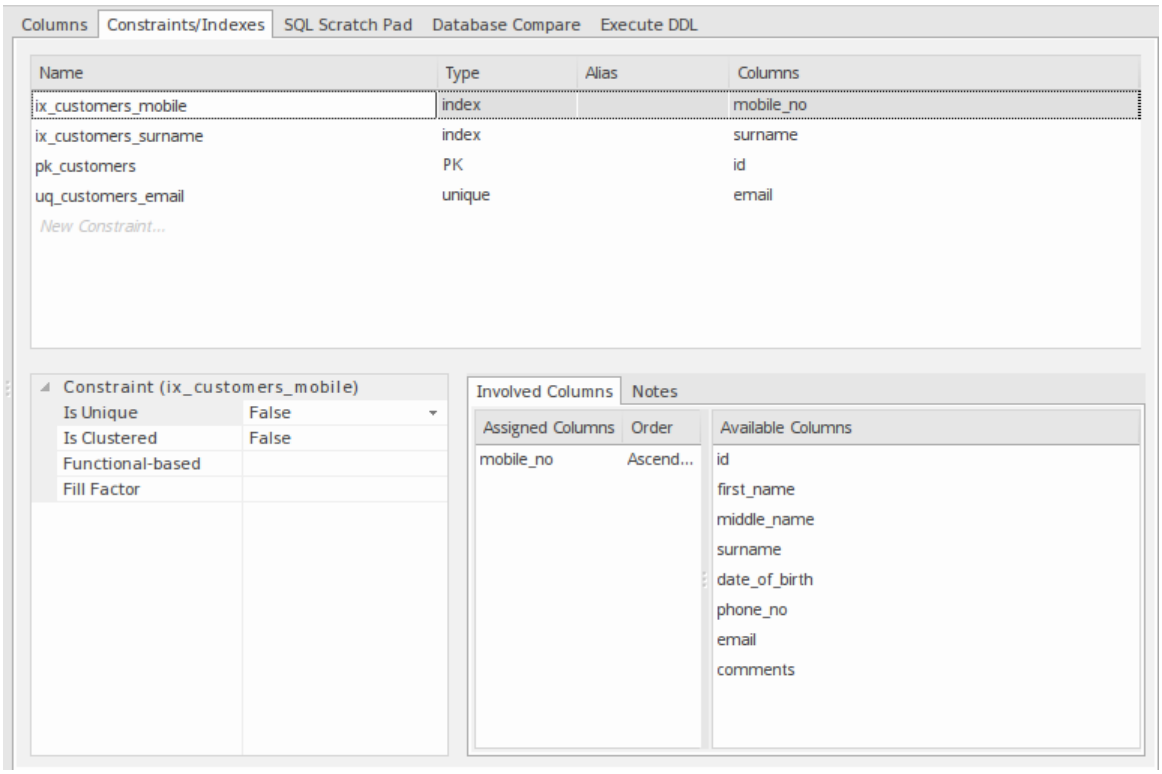
Dans Enterprise Architect un index est modélisé comme une opération stéréotypée.

Certains SGBD support des types d'index spéciaux ; Enterprise Architect les définit à l'aide de propriétés supplémentaires telles que basées sur les fonctions, clusterisées et facteurs de remplissage.

Accéder

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur le nom Tableau > Contraintes/Index
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Index
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index

Travailler sur un index



Étape	Action
1	Dans l'onglet 'Contraintes/Index' du Tableau , cliquez-droit et sélectionnez 'Ajouter une nouvelle

	contrainte'. La nouvelle contrainte est ajoutée avec le nom par défaut « contrainte1 » et le Type « index ». Remplacez le nom par votre nom d'index préféré.
2	Attribuez les colonnes appropriées à l'index. La liste « Colonnes affectées » comporte un champ « Ordre » supplémentaire qui spécifie l'ordre (croissant ou décroissant) dans lequel chaque colonne attribuée est stockée dans l'index. Vous pouvez modifier l'ordre de chaque colonne, selon vos besoins. De plus, pour les index MySQL, un champ « Len » sera visible dans lequel vous pourrez définir des index partiels ; c'est-à-dire un index qui utilise le premier nombre de caractères « n » d'un champ texte. Le champ « Len » ne prend que des valeurs numériques entières comprises entre 0 et la longueur définie de la colonne. Une valeur de 0 (qui est la valeur par défaut) indique que la colonne entière doit être indexée.
3	Dans le panneau 'Propriété', révision les paramètres des propriétés étendues définies pour le SGBD actuel.

Propriétés supplémentaires

Propriété	Description
Est unique	(Vrai / Faux) indique si l'index actuel est un 'Index Unique'. Un index unique garantit que la ou les colonnes indexées ne contiennent pas de valeurs en double, garantissant ainsi que chaque ligne a une valeur unique (ou une combinaison de valeurs lorsque l'index est constitué de plusieurs colonnes).
Est groupé	(Vrai / Faux) indique si l'index actuel est un « index clusterisé ». Avec un index clusterisé, les lignes du tableau sont physiquement stockées dans le même ordre que dans l'index, il ne peut donc y avoir qu'un seul index clusterisé par tableau . Par défaut, la Primary Key d'un tableau est clusterisée. Tous les SGBD ne prennent pas support les index clusterisés. Par conséquent, la propriété d'index « Est clusterisé » ne sera visible que pour les SGBD qui la support .
Est-ce qu'un bitmap	(Vrai / Faux) indique si l'index actuel est un index 'Bitmap'. Les index bitmap sont destinés à être utilisés sur des colonnes qui ont relativement peu de valeurs uniques (appelées colonnes à « faible cardinalité ») et qui sont physiquement constituées d'un tableau de bits (communément appelé bitmaps) pour chaque valeur unique. Chacun des tableaux aura un bit pour chaque ligne du tableau . Prenons cet exemple : un index bitmap est créé sur une colonne appelée « Gender », qui comporte les options « Mâle » ou « Femme ». Physiquement, l'index sera composé de deux tableaux de bits, un pour « Mâle » et un pour « Femme ». Le tableau de bits femelle aura un 1 dans chaque bit où la ligne correspondante a la valeur « Femme ». Les propriétés « Is Bitmap » et « Is Unique » s'excluent mutuellement et la génération DDL ignorera donc la propriété « Is Unique » lorsque la propriété « Is Bitmap » est True. Les index bitmap ne sont pris en charge que par Oracle ; par conséquent, cette propriété n'est visible que lors modélisation des index Oracle.
Facteur de remplissage	Une valeur numérique comprise entre 0 et 100, qui définit le pourcentage d'espace disponible qui doit être utilisé pour les données.

	Tous les SGBD ne support le facteur de remplissage, par conséquent la propriété d'index « Fill Factor » ne sera visible que pour les SGBD qui la support .
Basé sur le fonctionnel	Une instruction SQL qui définit la fonction/instruction qui sera évaluée et les résultats indexés ; Par exemple: INFÉRIEUR("champ") Tous les SGBD ne prennent pas support les index fonctionnels. Par conséquent, la propriété Index « Functional-based » ne sera visible que pour les SGBD qui les support , tels que PostgreSQL et Oracle.
Inclure	Identifie une liste CSV de colonnes non clés du tableau actuel. Tous les SGBD ne prennent pas support la propriété « Inclure » sur les index. Cette propriété ne sera donc visible que pour les SGBD qui la support .

Notes

- Avertissement : Enterprise Architect suppose qu'au moins une colonne est attribuée aux index ; cependant, Enterprise Architect n'applique pas cette règle lors de modélisation
  Si le DDL est généré pour un Tableau dont l'index est défini sans colonne(s) attribuée(s), ce DDL ne sera pas valide, sauf si l'index est basé sur les fonctionnalités.
- Toutes les colonnes affectées à un index fonctionnel sont ignorées

Contraintes uniques

Les contraintes uniques imposent le « caractère unique » d'un ensemble de champs dans toutes les lignes d'un Tableau , ce qui signifie que deux lignes d'un Tableau ne peuvent pas avoir les mêmes valeurs dans les champs d'une contrainte unique. Les contraintes uniques sont similaires aux Primary Keys (dans le sens où elles imposent également « l'unicité »), mais la principale différence est qu'un Tableau peut avoir plusieurs contraintes uniques définies mais une seule Primary Key .

Accéder

Ruban	Développer > Modélisation des données > Générateur de base de données > Cliquez sur le nom Tableau > Contraintes/Index > Cliquez-droit > Ajouter une nouvelle contrainte
Menu Contexte	Dans diagramme ou dans la fenêtre Navigateur | Cliquez-droit sur l'élément Tableau | Fonctionnalités | Contraintes/Index
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index : Ctrl+N

Créer une contrainte

Étape	Action
1	Dans l'onglet 'Contraintes/Index', une nouvelle contrainte est automatiquement créée et se voit attribuer le nom de contrainte par défaut et un ' Type ' d'index. Remplacez le nom de la contrainte par un nom qui l'identifie comme une contrainte unique.
2	Dans le champ ' Type ', changez la valeur de 'index' à 'unique'.

Notes

- Avertissement : Enterprise Architect suppose qu'au moins une colonne est affectée aux contraintes uniques ; cependant, Enterprise Architect n'applique pas cette règle lors de modélisation
  Si le DDL est généré pour un Tableau qui a une contrainte unique définie sans colonne(s) attribuée(s), ce DDL ne sera pas valide.

Foreign Keys

Une Foreign Key définit une colonne (ou une collection de colonnes) qui applique une relation entre deux Tableaux . Il est de la responsabilité du serveur de base de données de faire respecter cette relation afin de garantir l'intégrité des données. La définition de modèle d'une Foreign Key consiste en un Tableau parent (primaire) contenant un ensemble unique de données qui est ensuite référencé dans un Tableau enfant (étranger).

Dans Enterprise Architect , une Foreign Key est modélisée avec deux composants UML différents (mais liés) :

- Une contrainte Foreign Key (une opération UML avec le stéréotype de « FK ») stockée sur le Tableau enfant et
- Un connecteur Association (stéréotype de « FK ») définissant la relation entre les deux Tableaux

Créer une Foreign Key

Bien que la définition d'une Foreign Key puisse être complexe, l'écran Contrainte Foreign Key simplifie la modélisation des Foreign Keys . Cet écran est spécialement conçu pour vous aider à sélectionner la contrainte du Tableau parent à utiliser et fera automatiquement correspondre les colonnes Tableau enfants à celles du Tableau parent qui font partie de la contrainte. Différents aspects du processus de développement d'une Foreign Key sont décrits ici séparément à titre d'illustration, mais le processus global doit être une transition en douceur.

Un certain nombre de conditions doivent être remplies avant qu'une définition Foreign Key puisse être enregistrée :

- Les deux Tableaux doivent avoir des SGBD correspondants définis
- Le Tableau parent doit avoir au moins une colonne
- Le Tableau parent doit avoir une Primary Key , une contrainte unique ou un index unique défini

Créer une Foreign Key - à l'aide du générateur de base de données

Étape	Action
1	Dans l'arborescence Database Builder, cliquez-droit sur le nom Tableau enfant et cliquez sur 'Ajouter une nouvelle Foreign Key sur <nom tableau >'. Une dialogue s'affiche répertoriant tous les Tableaux parents possibles.
2	Double-cliquez sur le nom Tableau parent requis dans la liste ou sélectionnez-le et cliquez sur le bouton OK . L'écran « Contrainte Foreign Key » s'affiche.

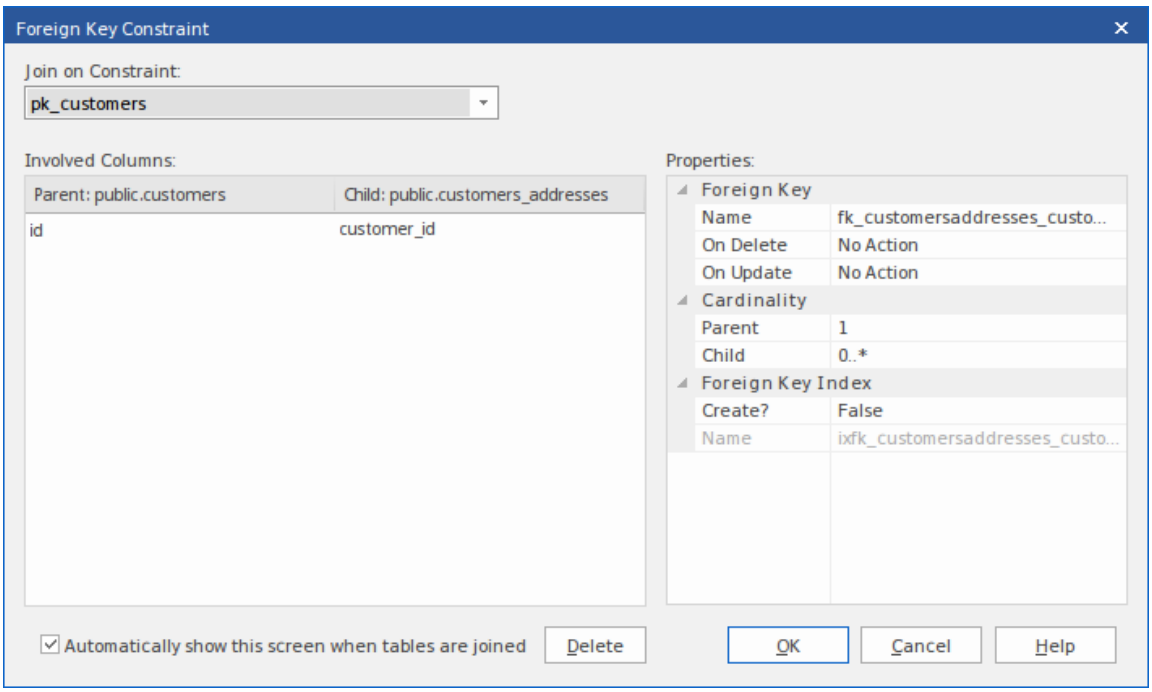
Créer une Foreign Key - en utilisant une relation sur un diagramme

Étape	Action
1	Dans le diagramme Modélisation des données, localisez le Tableau enfant (Foreign Key) et Tableau parent (Primary Key) requis.
2	Sélectionnez un connecteur Association dans la page ' Modélisation des données' de la Boîte à outils Diagramme .

3	Cliquez sur le Tableau enfant et dessinez le connecteur vers le Tableau parent.
4	Si l'écran Contrainte Foreign Key a été configuré pour s'afficher automatiquement lorsque deux Tableaux sont joints, il affiche maintenant . Sinon, soit : • Double-cliquez sur le connecteur ou • Cliquez-droit sur le connecteur et sélectionnez l'option ' Foreign Keys ' L'écran Contrainte Foreign Key s'affiche.

L'écran de contrainte Foreign Key

À titre d'exemple, cette image montre l'écran de contrainte Foreign Key chargé avec les détails de « fk_customersaddresses_customers » (tel que défini dans l'exemple de modèle).

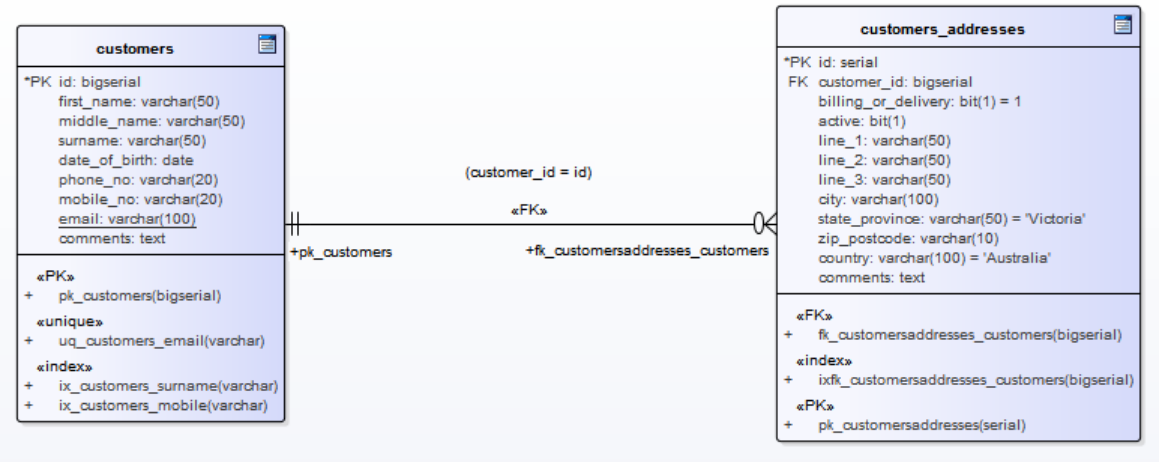


Option	Action
Rejoindre sur contrainte	Cette zone de liste déroulante répertorie toutes les contraintes définies dans le Tableau parent qui pourraient être utilisées comme base d'une Foreign Key . (Ces contraintes peuvent être Primary Keys , des contraintes uniques ou des index uniques.) La première contrainte de la liste est sélectionnée par défaut ; si ce n'est pas la contrainte souhaitée, sélectionnez la contrainte correcte dans la zone de liste déroulante. Lorsque vous sélectionnez la contrainte, ses colonnes sont automatiquement répertoriées dans le panneau « Colonnes impliquées », sous la colonne « Parent : <tablename> ».
Colonnes impliquées	Cette liste est divisée en deux : les colonnes impliquées dans la contrainte sélectionnée sont répertoriées à gauche, et les colonnes enfants qui vont être appariées aux colonnes parents sont répertoriées à droite. Lorsqu'une contrainte est sélectionnée (dans le champ 'Joindre sur contrainte') le côté parent est actualisé pour afficher toutes les colonnes affectées à la contrainte

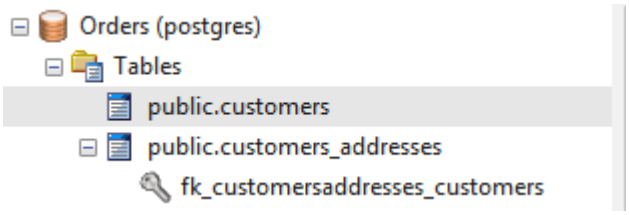
	sélectionnée. Du côté enfant, le système tentera automatiquement de faire correspondre chaque colonne parent à une colonne du même nom dans le Tableau enfant. Si le Tableau enfant n'a pas de colonne du même nom, une nouvelle colonne de ce nom sera ajoutée à la liste, marquée par (*) pour indiquer qu'une nouvelle colonne sera créée dans Tableau . Cependant, si vous souhaitez forcer le couplage avec une colonne Tableau enfant existante ou une nouvelle colonne avec un nom différent, cliquez sur le champ du nom de la colonne et soit : • Type le nom de remplacement, ou • Sélectionnez une colonne existante (cliquez sur la flèche déroulante et sélectionnez le nom dans la liste)
Nom	Ce champ définit le nom de la contrainte Foreign Key et prend par défaut un nom construit par le nom Foreign Key Gabarit . Pour changer le nom en quelque chose d'autre que le nom par défaut, écrasez simplement la valeur .
Lors de la suppression	Sélectionnez l'action qui doit être effectuée sur les données du Tableau enfant lorsque les données du Tableau parent sont supprimées, afin de maintenir l'intégrité référentielle.
En mise à jour	Sélectionnez l'action à entreprendre sur les données du Tableau enfant lorsque les données du Tableau parent sont mises à jour, afin de maintenir l'intégrité référentielle.
Parent	Cliquez sur la flèche déroulante et sélectionnez la cardinalité du Tableau parent dans la Foreign Key .
Enfant	Cliquez sur la flèche déroulante et sélectionnez la cardinalité du Tableau enfant dans la Foreign Key .
Créer?	Si vous souhaitez créer un index Foreign Key en même temps que la Foreign Key , définissez cette propriété sur True. Le nom de l'index Foreign Key est contrôlé par le gabarit de l'index Foreign Key et le nom généré est affiché dans le champ « Nom » sous le champ « Créer ? champ.
Afficher automatiquement cet écran lorsque tableaux sont joints	(Pour modélisation schématique) Cochez cette case pour afficher automatiquement cet écran à chaque fois qu'une Association est créée entre deux Tableaux .
Supprimer	Cliquez sur ce bouton pour supprimer la Foreign Key existante (enregistrée) actuellement sélectionnée. Une prompt s'affiche pour confirmer la suppression (et la suppression de l'index Foreign Key , s'il existe) - cliquez sur le bouton Oui. La suppression d'une Foreign Key laisse en place un connecteur d'association, que vous pouvez modifier ou supprimer (cliquez-droit et sélectionnez 'Supprimer l'association : à < Nom Tableau >').
OK	Cliquez sur ce bouton pour enregistrer la Foreign Key .

Exemples

Cet exemple montre Foreign Keys simples dans un diagramme :



La même Foreign Key sera affichée dans l'arborescence du générateur de base de données en tant que nœud enfant sous « customer.addresses » de Tableau .



Vérifier les contraintes

Une contrainte de vérification renforce l'intégrité du domaine en limitant les valeurs acceptées par une colonne.

Accéder

Ruban	Développer > Modélisation des données > Générateur de base de données > Cliquez sur le nom Tableau > Contraintes/Index > Cliquez-droit > Ajouter une nouvelle contrainte
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Index | Cliquez-droit | Ajouter une nouvelle contrainte
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index : Ctrl+N

Créer une contrainte

Étape	Action
1	Dans l'onglet « Contraintes/Index » de l'écran Colonnes et contraintes, une nouvelle contrainte est automatiquement créée et se voit attribuer le nom de contrainte par défaut et un « Type » d'index. Remplacez le nom de la contrainte par un nom qui identifie la contrainte en tant que contrainte de vérification, tel que « CHK_ColumnName » (le préfixe CHK_ est facultatif).
2	Dans le champ ' Type ', changez la valeur de 'index' à 'check'.
3	Dans le panneau ' Propriétés ' de la propriété Condition, tapez l'instruction SQL qui sera utilisée comme condition de vérification ; par exemple, colonne1 < 1 000. Si la condition est longue, cliquez sur le bouton pour afficher un éditeur SQL (avec coloration syntaxique).

Supprimer une contrainte de vérification

Si vous ne souhaitez pas conserver de contrainte de contrôle, soit :

- Cliquez-droit dessus dans la liste et sélectionnez 'Supprimer la contrainte <nom>', ou
- Cliquez sur l'élément et appuyez sur Ctrl+D

La contrainte est immédiatement supprimée.

Notes

- Toutes les colonnes affectées à une contrainte de vérification sont ignorées

Tableau Déclencheurs

Un déclencheur Tableau est du SQL ou du code qui est automatiquement exécuté suite à la modification de données dans une base de données Tableau . Déclencheurs sont hautement personnalisables et peuvent être utilisés de différentes manières ; par exemple, ils pourraient être utilisés pour empêcher l'exécution de certaines activités de base de données pendant les heures de bureau, ou pour fournir une validation ou effectuer des suppressions dans les Tableaux secondaires lorsqu'un enregistrement dans le Tableau principal est supprimé.

Dans Enterprise Architect , un déclencheur Tableau est modélisé comme une opération stéréotypée et géré à l'aide de l'écran « Contraintes » de Tableau .

Accéder

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur le nom Tableau > Contraintes/Index | Cliquez-droit | Ajouter une nouvelle contrainte
Menu Contexte	Dans diagramme | Cliquez-droit sur Tableau | Fonctionnalités | Contraintes/Indices | Cliquez-droit | Ajouter une nouvelle contrainte
Raccourcis Clavier	Cliquez sur Tableau : F9 > Contraintes/Index : Ctrl+N

Créer un Tableau Déclencheur

Étape	Action
1	Dans l'onglet 'Contraintes/Index', une nouvelle contrainte est automatiquement créée et se voit attribuer le nom de contrainte par défaut et un ' Type ' d'index. Remplacez le nom de la contrainte par un nom qui identifie la contrainte comme un déclencheur , tel que TRG_OnCustomerUpdate. (Le préfixe TRG_ est facultatif.)
2	Dans le champ ' Type ', changez la valeur de 'index' à ' déclencheur '.
3	Dans le panneau ' Propriétés ' de la propriété Statement, saisissez l'instruction SQL complète (y compris CREATE TRIGGER) qui définira le Déclencheur . Si la condition est longue, cliquez sur le bouton pour afficher un éditeur SQL (avec coloration syntaxique).
4	Les propriétés Déclencheur Time et Déclencheur Event sont actuellement des valeurs à titre informatif et ne sont pas utilisées dans la génération DDL.

Supprimer un Tableau Déclencheur

Si vous ne souhaitez pas conserver de déclencheur , soit :

- Cliquez-droit dessus dans la liste et sélectionnez 'Supprimer la contrainte <nom>', ou

- Cliquez sur l'élément et appuyez sur Ctrl+D

Le déclencheur est immédiatement supprimé.

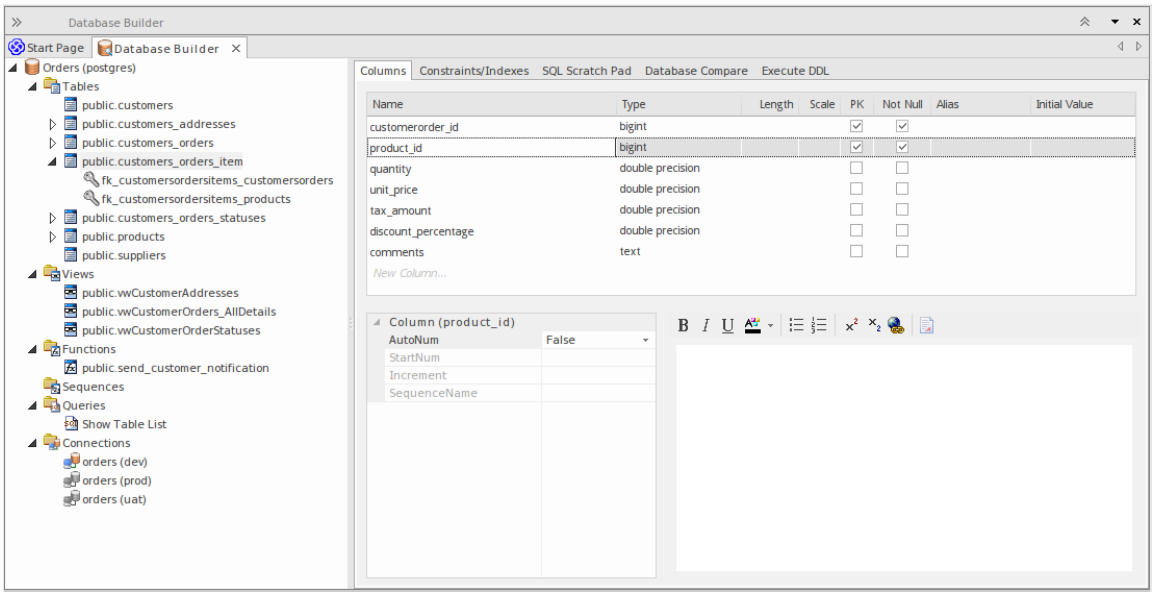
Notes

- Toutes les colonnes affectées au tableau déclencheurs sont ignorées

Colonnes

Tableaux sont l' object fondamental de la base de données, et les colonnes (et leurs propriétés) sont les fonctionnalité Tableau les plus fréquemment modifiées et modifiées par les modélisateurs de données. Par conséquent, la page « Colonnes » est idéalement située en tant que première page de l'interface du générateur de base de données.

Une fois qu'un Tableau est sélectionné dans l'arborescence de Database Builder, la page « Colonnes » est remplie avec la liste de colonnes actuellement définie pour ce Tableau . Le modélisateur de données peut ensuite apporter des modifications aux propriétés de la colonne principale directement dans la liste ou la grille. Lorsque le modélisateur de données sélectionne des colonnes individuelles dans la liste, les propriétés étendues (et les commentaires) de la colonne sont affichées immédiatement sous la liste, permettant ainsi la modification de ces propriétés étendues.



Cette figure montre l'interface Database Builder affichant l'arborescence des objets et l'onglet Colonnes affichant les colonnes du tableau sélectionné.

Notes

- La page « Colonnes » ne sera remplie que lorsqu'un élément Tableau est sélectionné dans l'arborescence du générateur de base de données.

Créer des colonnes Tableau de base de données

Une colonne de base de données Tableau est représentée dans le profil Modélisation de données UML comme un attribut avec le stéréotype « colonne ». Pour un Tableau sélectionné, vous pouvez révision les colonnes existantes et créer de nouvelles colonnes, sur la page « Colonnes » du générateur de base de données ou sur l'écran « Colonnes et contraintes ».

Vous pouvez définir les détails des colonnes directement sur la liste des colonnes dans l'onglet 'Colonnes'. Les modifications sont automatiquement enregistrées au fur et à mesure que vous remplissez chaque champ. Certains champs comportent certaines restrictions quant aux données que vous pouvez saisir, comme décrit ici. L'onglet contient également un panneau ' Propriétés ' et un champ ' Notes ', qui sont renseignés avec les informations existantes sur la colonne sélectionnée. Chaque nouvelle colonne que vous créez se voit automatiquement attribuer un ensemble de valeurs par défaut et est ajoutée au bas de la liste.

Accéder

Ruban	Développer > Modélisation de données > Générateur de base de données > Cliquez sur Tableau > Colonnes > Clic droit > Ajouter une nouvelle colonne
Menu Contexte	Dans diagramme , cliquez-droit sur le Tableau | Fonctionnalités | Colonnes | Clic droit | Ajouter une nouvelle colonne
Raccourcis Clavier	Sélectionnez un tableau | F9 | Touche de tabulation (pour définir le focus de saisie sur l'onglet « Colonnes ») | Ctrl+N

Créer des colonnes dans un Tableau

Option	Action
Nom	Remplacez le nom par défaut par le texte du nom de colonne approprié.
Type	Cliquez sur la flèche déroulante et sélectionnez le type de données approprié pour la colonne. Les types de données disponibles dépendent du SGBD attribué au Tableau parent.
Longueur	(Facultatif) Certains types de données ont un composant de longueur. Par exemple, VARCHAR a une longueur qui définit le nombre de caractères pouvant être stockés. Si le type de données n'a pas de composant de longueur, ce champ est désactivé. Si le champ est disponible et si vous devez définir un nombre de caractères, saisissez ici la valeur .
Échelle	(Facultatif) Certains types de données ont un composant d'échelle. Par exemple, DECIMAL a une échelle qui définit le nombre de décimales pouvant être conservées. Si le type de données n'a pas de composant d'échelle, ce champ est désactivé. Si le champ est disponible et si vous devez définir une échelle, saisissez la valeur ici.

PK	Cochez la case si la colonne fait partie de la Primary Key de ce Tableau .
Pas nul	Cochez la case si les valeurs vides sont interdites pour cette colonne. La case à cocher est désactivée si la case « PK » est cochée.
Alias	Si cela est nécessaire à des fins d'affichage et de documentation, saisissez un autre nom pour le champ.
Valeur initiale	Si nécessaire, saisissez une valeur qui peut être utilisée comme valeur par défaut pour cette colonne.
Notes	Type toute information supplémentaire nécessaire pour documenter la colonne. Vous pouvez formater le texte à l'aide de la barre d'outils Notes en haut du champ.

Propriétés des colonnes

Les propriétés appropriées pour le système de gestion de base de données de Tableau s'affichent automatiquement dans le panneau « Propriétés » (développez la branche « Colonne (<nom>) » si elles ne sont pas visibles).

Propriété	SGBD
Autonum (Incrément de Startnum)	Oracle MySQL serveur SQL DB2 PostgreSQL Notes : Si vous avez besoin d'une séquence de numérotation automatique, définissez cette propriété sur True et, si nécessaire, définissez le numéro de départ et l'incrément.
Généré	DB2 Notes : Définissez cette propriété supplémentaire pour la numérotation automatique dans DB2 sur « Par défaut » ou « Toujours ».
PasPourRep	Serveur SQL Notes : Définissez cette propriété sur True si vous souhaitez bloquer la réplication.
Remplissage zéro	MySQL Notes : Définissez cette propriété sur True ou False pour indiquer si les champs sont remplis à zéro ou non.
Non signé	MySQL Notes : Définissez cette propriété sur True ou False pour indiquer si les champs acceptent ou non les nombres non signés.
Type de longueur	Oracle Notes : Définissez cette propriété pour définir la sémantique des caractères comme 'Aucun', 'Byte' ou 'Char'.

Supprimer les colonnes Tableau de la base de données

Pour une base de données Tableau sélectionnée, vous pouvez révision les colonnes existantes et supprimer n'importe quelle colonne individuelle, dans l'onglet « Colonnes » de l'écran Colonnes et contraintes.

Accéder

Utilisez l'une des méthodes décrites ici pour afficher une liste de colonnes pour un tableau , puis sélectionnez une colonne et supprimez-la.

Lorsque vous sélectionnez l'option 'Supprimer la colonne '<nom>", si toutes les règles de validation sont satisfaites, la colonne est immédiatement supprimée.

Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur Tableau > Colonnes > Cliquez-droit sur le nom de la colonne > Supprimer la colonne <nom>
Menu Contexte	Dans diagramme , cliquez-droit sur le Tableau | Fonctionnalités | Colonnes | Cliquez-droit sur le nom de la colonne | Supprimer la colonne <nom>
Raccourcis Clavier	F9 | Utilisez « Flèche vers le haut » ou « Flèche vers le bas » pour sélectionner une colonne | Ctrl+D

Notes

- Si la colonne Tableau de base de données supprimée est impliquée dans des contraintes, elle en sera automatiquement supprimée

Réorganiser les colonnes Tableau de la base de données

Si vous avez plusieurs colonnes définies dans une base de données Tableau , vous pouvez modifier l'ordre dans lequel elles sont répertoriées. L'ordre dans la liste est l'ordre dans lequel les colonnes apparaissent dans le DDL généré.

Accéder

Utilisez l'une des méthodes décrites ici pour afficher une liste de colonnes pour un Tableau , puis sélectionnez une colonne et repositionnez-la dans la liste.

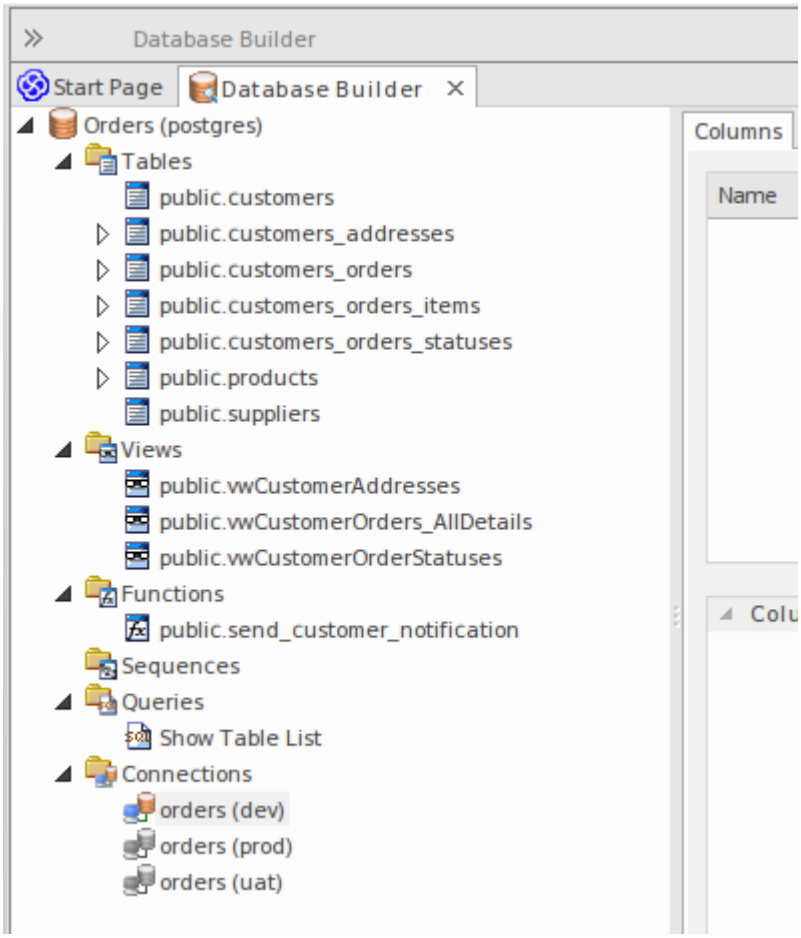
Ruban	Développer > Modélisation des données > Database Builder > Cliquez sur Tableau
Menu Contexte	Dans diagramme , cliquez-droit sur le Tableau | Fonctionnalités | Colonnes
Raccourcis Clavier	F9

Changer l'ordre des colonnes

Étape	Action
1	Dans l'onglet « Colonnes », cliquez sur le nom de la colonne souhaitée dans la liste.
2	Cliquez-droit et sélectionnez le : • Option « Déplacer la colonne <nom> vers le haut » (ou appuyez sur Ctrl+Flèche vers le haut) pour déplacer la colonne d'une position vers le haut. • Option « Déplacer la colonne <nom> vers le bas » (ou appuyez sur Ctrl+Flèche vers le bas) pour déplacer la colonne d'une position vers le bas. Ces options ont un effet immédiat aussi bien dans l'onglet 'Colonnes' que sur un diagramme .

Travailler dans le générateur de base de données

Lorsqu'un modèle de données est chargé, le générateur de base de données crée un ensemble de dossiers logiques, un pour chaque type object pris en charge par le SGBD actuel. Chaque dossier logique est rempli avec tous les objets de ce type trouvés dans la hiérarchie des Paquetages du modèle de données (comme indiqué dans la fenêtre Navigateur).



Dans cette image, le modèle de données « Commandes (postgres) » montre les dossiers logiques pour Tableaux , Vues , les Fonctions, les Séquences, les Requêtes et les Connexions. Il convient de noter qu'il n'existe pas de dossier pour « Procédures » puisque PostgreSQL ne prend pas support les procédures de base de données.

Actions disponibles dans l'arborescence du générateur de base de données

La majorité des fonctions de Database Builder sont accessibles via des menus contextuels. Chaque object de l'arborescence possède son propre ensemble d'éléments de menu uniques en fonction de son type et de son statut. Ce tableau décrit les éléments de menu contextuel disponibles et identifie les objets auxquels ils s'appliquent.

Option de menu	S'applique à / Description
Nouveau modèle de données	*S'applique à :* Espace vide *Description :* Ouvre l'onglet Page Démarrage 'Créer à partir de Motif ' (Assistant de Modèle).
Rafraîchir tout	*S'applique à :* Espace vide *Description :* Recharge la liste complète des modèles de données.

Charger	*S'applique à :* nœud racine *Description :* charge tous les détails du modèle de données.
Décharger	*S'applique à :* nœud racine *Description :* Décharge tous les détails du modèle de données.
Importer un schéma de base de données	*S'applique à :* nœud racine chargé *Description :* Ouvre la dialogue « Importer un schéma de base de données » en utilisant la connexion active actuelle comme source de base de données Live.
Générer DDL	*S'applique à :* Nœud racine chargé, Dossier, Tableau , Vue , Procédure, Fonction, Séquence , Paquetage *Description :* Ouvre la dialogue ' Générer DDL' avec le ou les object actuels sélectionnés.
Afficher les différences	*S'applique à :* nœud racine chargé, dossier, Tableau , Vue , procédure, fonction, Séquence *Description :* Compare les objets sélectionnés à la connexion active actuelle.
Afficher les différences avec les options	*S'applique à :* Nœud racine chargé, Dossier, Tableau , Vue , Procédure, Fonction, Séquence , Paquetage *Description :* compare les objets sélectionnés à la connexion active actuelle et ignore éventuellement certaines des différences en fonction des options de comparaison spécifiées.
Gérer les options du SGBD	*S'applique à :* nœud racine chargé *Description :* Ouvre la dialogue « Gérer les options du SGBD », qui peut être utilisée pour modifier le SGBD alloué et le propriétaire de plusieurs objets.
Nombre d'enregistrements Vue	*S'applique à :* Tableau , Vue *Description :* crée et exécute une requête SELECT (formatée pour s'adapter au SGBD de l'élément) pour afficher le nombre d'enregistrements dans le Tableau ou Vue sélectionné. S'il n'y a aucune connexion active, vous êtes invité à en sélectionner une.
Vue des 100 premières lignes	*S'applique à :* Tableau , Vue *Description :* crée et exécute une requête SELECT (formatée pour s'adapter au SGBD de l'élément) pour afficher les 100 premières lignes du Tableau ou Vue sélectionné. S'il n'y a aucune connexion active, vous êtes invité à en sélectionner une.
Vue des 1 000 premières lignes	*S'applique à :* Tableau , Vue *Description :* crée et exécute une requête SELECT (formatée pour s'adapter au SGBD de l'élément) pour afficher les 1 000 premières lignes du Tableau ou Vue sélectionné. S'il n'y a aucune connexion active, vous êtes invité à en sélectionner une.
Vue toutes les lignes	*S'applique à :* Tableau , Vue *Description :* crée et exécute une requête SELECT (formatée pour s'adapter au SGBD de l'élément) pour afficher toutes les lignes du Tableau ou Vue sélectionné. S'il n'y a aucune connexion active, vous êtes invité à en sélectionner une.

Propriétés	*S'applique à :* Nœud racine chargé, Dossier, Tableau , Vue , Procédure, Fonction, Séquence , Paquetage , Connexion *Description :* Ouvre la dialogue standard ' Propriétés ' pour l' object sélectionné.
Rechercher dans Projet Navigateur	*S'applique à :* Nœud racine chargé, Dossier, Tableau , Vue , Procédure, Fonction, Séquence , Paquetage , Query SQL, Connexion *Description :* Recherche l' object sélectionné dans la fenêtre Navigateur .
Rafraîchir	*S'applique à :* nœud racine chargé *Description :* recharge les détails du modèle de données actuellement chargé. Cela est nécessaire lorsque des objets sont ajoutés, modifiés ou supprimés par d'autres utilisateurs ou lorsque les modifications sont effectuées en dehors du générateur de base de données.
Ajouter un nouveau <type>	*S'applique à :* Dossier, Tableau , Vue , Procédure, Fonction, Séquence , Paquetage , Query SQL, Connexion *Description :* Crée un nouvel object du type spécifié.
Cloner <nom>	*S'applique à :* Dossier, Tableau , Vue , Procédure, Fonction, Séquence , Paquetage , Query SQL, Connexion *Description :* Effectue une nouvelle copie de l' object sélectionné. Lorsque vous sélectionnez cette option, une prompt s'affiche dans laquelle vous définissez le nom et le propriétaire du nouvel object . Pour les objets Tableau , vous pouvez choisir quelles contraintes existantes doivent être copiées (et définir un nom pour chacune d'entre elles) ainsi que les Foreign Keys à copier. Pour les objets basés sur SQL, vous pouvez apporter toutes les modifications nécessaires au code SQL du nouvel élément.
Supprimer <nom>	*S'applique à :* Tableau , Vue , Procédure, Fonction, Séquence , Paquetage , Query SQL , Connexion *Description :* Supprime définitivement l' object sélectionné du référentiel.
Ajouter une nouvelle Foreign Key sur <nom>	*S'applique à :* Tableau *Description :* crée une nouvelle relation entre le Tableau sélectionné et un autre, puis affiche l'écran « Contrainte Foreign Key » pour la nouvelle relation.
Propriétés Object SQL	*S'applique à :* Vue , Procédure, Fonction, Séquence *Description :* Ouvre l'écran "Éditeur Object SQL".
Modifier	*S'applique à :* Query SQL *Description :* charge le code SQL (tel que défini dans l'élément sélectionné) dans SQL Scratch Pad.
Exécuter	*S'applique à :* Query SQL *Description :* charge le code SQL dans SQL Scratch Pad et l'exécute. S'il n'y a aucune connexion active, vous êtes invité à en sélectionner une.
Définir comme connexion de base de données active	*S'applique à :* Connexion *Description :* marque la connexion à la base de données sélectionnée comme étant active pour la session en cours.

Définir comme connexion de base de données par défaut	*S'applique à :* Connexion *Description :* marque la connexion à la base de données sélectionnée comme étant active à chaque fois que le modèle de données est chargé.
Propriétés de connexion à la base de données	*S'applique à :* Connexion *Description :* Ouvre l'écran ' Propriétés de connexion à la base de données', pour gérer les paramètres de connexion.

Créer/Modifier/Supprimer des objets de base de données

Les pages répertoriées dans cette section décrivent en détail comment utiliser l'interface de Database Builder pour créer et manipuler Tableaux de base de données ; cependant, le processus de création et de manipulation des objets de base de données SQL est documenté dans d'autres domaines. Consultez ces sujets pour plus de détails :

- Database Views
- Database Procedures
- Database Functions
- Database Sequences
- Database Connections

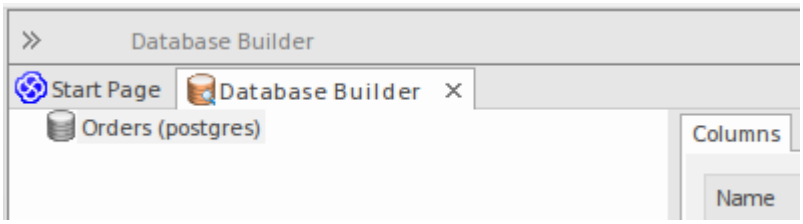
Connexions à la base de données dans le générateur de base de données

Lors de l'exécution de certaines tâches telles que « Comparer » ou « Exécuter DDL », le générateur de base de données nécessite une connexion à la base de données active. Une seule connexion à la base de données peut être rendue active (indiquée par une icône colorée « Connexion à la base de données », tandis que les autres sont grises) à un instant donné. Si une connexion à la base de données n'est pas actuellement active et que vous essayez d'effectuer une tâche qui en nécessite une, le générateur de base de données effectue l'une de ces actions en fonction du nombre de connexions définies :

- 0 Connexions – vous prompts à créer une connexion et, en cas de succès, continue
- 1 Connexion – la définit comme active et continue
- 2 (ou plus) connexions – vous prompts à en sélectionner une et, en cas de succès, continue

Ouverture du générateur de base de données

Lorsque vous ouvrez le générateur de base de données pour la première fois, il recherche dans l'ensemble du projet tous Paquetages ayant le stéréotype « Modèle de données » et charge les modèles de données correspondants en tant que nœuds racine dans l'arborescence. Une icône grisée indique que les détails du modèle de données ne sont pas chargés.



Cette figure montre le générateur de base de données avec un seul modèle de données déchargé appelé « Commandes (postgres) ».

Utilisation du générateur de base de données

Vous pouvez commencer à travailler dans Database Builder de l'une des deux manières suivantes :

Tâche	Action
Créer un nouveau modèle de données	Une fois que vous avez ouvert la vue Database Builder, cliquez-droit dans l'espace vide de l'arborescence et sélectionnez 'New Data Modèle ' pour invoquer l' Assistant de Modèle (onglet Page Démarrage 'Create from Motif ').
Charger un Modèle de données existant	Une fois la vue Database Builder ouverte, chargez l'un des modèles de données définis de l'une des manières suivantes : • Faites un clic droit sur le nom et sélectionnez « Charger », ou • Double-cliquez sur le nom

Propriétés Modèle de données

Dans les versions antérieures d' Enterprise Architect (avant l'introduction de Database Builder), il était nécessaire que le modélisateur de données définisse manuellement les propriétés des objets de base de données avant que certaines tâches soient autorisées. Par exemple, Enterprise Architect ne permettrait pas la définition d'une colonne Tableau sans que Tableau ne reçoive au préalable un SGBD. En effet, le SGBD contrôle la liste des types de données disponibles.

Pour améliorer l'efficacité et l'expérience utilisateur, Database Builder définit des valeurs par défaut pour un certain nombre de propriétés au niveau du modèle de données, puis applique automatiquement ces valeurs par défaut chaque fois que de nouveaux objets sont créés.

Propriétés

Option	Description
SGBD	*Défini par rapport à :* Le Paquetage <<Database>> du modèle de données

	Défini comme : Valeur Étiquetée *Détails :* définit le SGBD du modèle de données actuel *Informations supplémentaires:* • Contrôle quels dossiers logiques sont affichés pour le modèle de données actuel dans l'arborescence du générateur de base de données. • Contrôle quelles règles de SGBD sont appliquées lors des comparaisons de bases de données • Est automatiquement attribué à chaque nouvel object de base de données créé dans le modèle de données actuel
Propriétaire par défaut	*Défini par rapport à :* Le Paquetage <<Database>> du modèle de données *Défini comme :* Valeur Étiquetée *Détails :* définit le propriétaire par défaut pour le modèle de données actuel. *Informations supplémentaires:* • Est automatiquement attribué à chaque nouvel object de base de données créé dans le modèle de données actuel, si le SGBD supporte les propriétaires/schémas
Connexion par défaut	*Défini par rapport à :* Le Paquetage <<Database>> du modèle de données *Défini comme :* Valeur Étiquetée *Détails :* (Facultatif) le nom de la connexion par défaut *Informations supplémentaires:* • Chaque fois qu'un modèle de données est chargé, la propriété « DefaultConnection » est vérifiée ; si elle est présente, la connexion portant ce nom est automatiquement rendue active • Le modèle d'ingénierie de base de données Motifs ne définit pas de valeur pour cette propriété, il est créé ou mis à jour chaque fois qu'un utilisateur définit une connexion comme valeur par défaut.

Notes

- Si un modèle de données est sélectionné dans la fenêtre Navigateur lors de l'ouverture du générateur de base de données, les détails de ce modèle seront automatiquement chargés.

Bloc-notes SQL

Le SQL Scratch Pad fournit un mécanisme permettant de développer et exécuter des requêtes SQL ad hoc sur une base de données active. Pendant que vous développez votre modèle de données, vous souhaiterez peut-être exécuter et tester des requêtes SQL ad hoc pour un script DDL, ou exécuter des requêtes sur la base de données en direct ; tout cela est possible dans l'interface Enterprise Architect Database Builder.

Le SQL Scratch Pad nécessite que le générateur de base de données dispose d'une connexion valide à une base de données active. Cette connexion à la base de données est partagée entre les onglets « SQL Scratch Pad », « Database Compare » et « Execute DDL » du Database Builder.

Le Scratch Pad se compose de :

- Une barre d'outils fournissant facilités pour importer, sauvegarder, exécuter et effacer les requêtes SQL
- Un panneau d'édition dans lequel vous créez ou importez les requêtes SQL. Ce panneau fournit une coloration syntaxique basée sur SQL pour le modèle de données actuel.
- Un panneau à onglets composé de deux pages, une pour afficher les résultats de l'exécution de la Query et une pour afficher les messages générés lors de l'exécution.

Accéder

Ouvrez la fenêtre Database Builder, puis affichez l'onglet « SQL Scratch Pad ».

Ruban	Développement > Modélisation de données > Générateur de base de données > SQL Scratch Pad

La barre d'outils du Bloc-notes

La fonctionnalité de chaque bouton de la barre d'outils Scratch Pad est décrite dans ce tableau , de gauche à droite.

Bouton	Action
Exécuter SQL	Exécute la Query SQL actuellement affichée dans le Bloc-notes. Vérifiez les onglets « Résultats » et « Messages » pour connaître le résultat de l'exécution de la Query .
Nouveau	Efface les champs de l'éditeur Query SQL afin que vous puissiez saisir une nouvelle requête.
Ouvrir	Charge une Query SQL à partir d'un fichier. Un navigateur de fichiers source s'affiche, par défaut pour afficher les fichiers SQL. Cliquez sur le nom du fichier et sur le bouton Ouvrir pour afficher le contenu du fichier dans le Scratch Pad.
Enregistrer dans Query SQL	Enregistre cette instruction SQL dans l' object Query SQL d'où elle provient.
Enregistrer dans une nouvelle Query SQL	Crée un nouvel object Query SQL et enregistre cette instruction dans cet object .

Enregistrer dans un fichier	Enregistre la Query actuellement affichée dans le fichier d'où elle provient. Si vous avez créé la Query à partir de zéro, un navigateur de fichiers source s'affiche dans lequel vous tapez le nouveau nom de fichier et cliquez sur le bouton Enregistrer pour enregistrer la Query .
Enregistrer dans un nouveau fichier	Enregistre la Query actuellement affichée dans un nouveau fichier .sql. Un navigateur de fichiers source s'affiche sur lequel vous saisissez le nouveau nom de fichier et cliquez sur le bouton Enregistrer pour enregistrer la Query .
Effacer	Efface le contenu du Bloc-notes. Toute Query affichée dans le Bloc-notes y reste jusqu'à ce que vous la remplaciez par une autre Query du fichier ou que vous fermiez le modèle.
Basculer le commentaire	Applique les caractères de commentaire SQL « -- » au début de chaque ligne sélectionnée ou, si les lignes sélectionnées sont déjà commentées, supprime les caractères de commentaire. Vous pouvez également appuyer sur Ctrl+Maj+C.
Séparateur d'instructions	Type le(s) caractère(s) à utiliser pour marquer la fin de chaque instruction.
Aide	Affiche l'aide sur le Bloc-notes Query SQL.
Description Query	Affiche une étiquette fournissant une description du SQL actuel, s'il y a des modifications en attente (indiquées par un *) et le nom de l' object Query SQL chargé ou du nom de fichier.

Notes

- Le SQL Scratch Pad ne manipule en aucun cas votre SQL, vous devez donc utiliser la syntaxe correcte pour le SGBD actuel.
- Bien que SQL Scratch Pad puisse exécuter plusieurs instructions SQL et que l'état et les messages de chaque instruction soient affichés dans la liste « Messages », seuls les résultats d'une instruction SELECT peuvent être affichés dans la liste « Résultats » à la fois ; toutes les instructions SELECT suivantes seront ignorées

Exécuter le DDL

L'onglet « Exécuter DDL » fournit un mécanisme permettant d'exécuter facilement les instructions DDL générées sur une base de données en direct et fournit un retour instantané sur leur réussite, le tout dans l'interface Enterprise Architect et sans avoir besoin d'autres produits.

Il existe deux types différents d'instructions DDL qu'Enterprise Enterprise Architect peut générer et envoyer à l'onglet « Exécuter DDL » :

- Créer des instructions DDL, créées par l'écran Générer DDL, et
- Modifier les instructions DDL, créées par la fenêtre Comparaison de bases de données

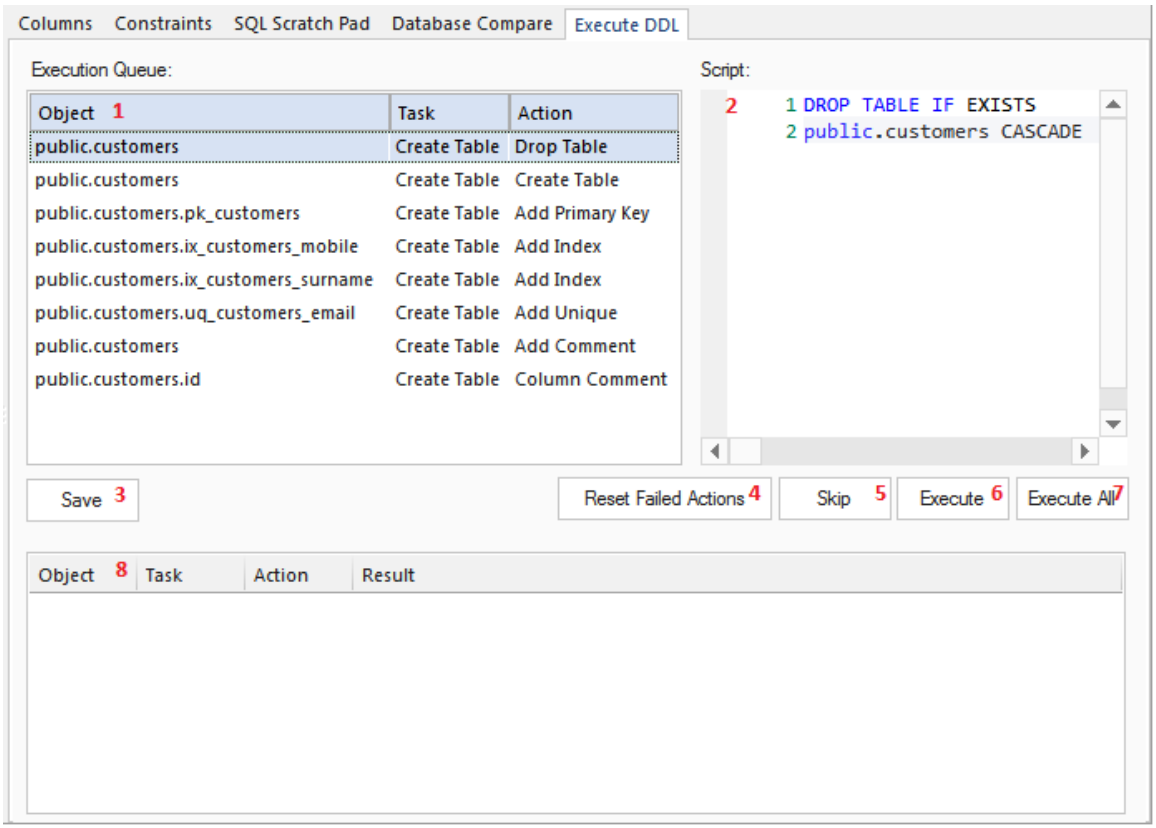
La fonctionnalité Exécuter DDL nécessite que Database Builder dispose d'une connexion valide à une base de données active. Cette connexion à la base de données est partagée entre les onglets SQL Scratch Pad, Database Compare et « Execute DDL » du Database Builder.

Accéder

Ouvrez la fenêtre Database Builder, puis affichez l'onglet 'Execute DDL'.

Ruban	Développer > Modélisation de données > Générateur de base de données > Exécuter DDL

Exécuter le DDL

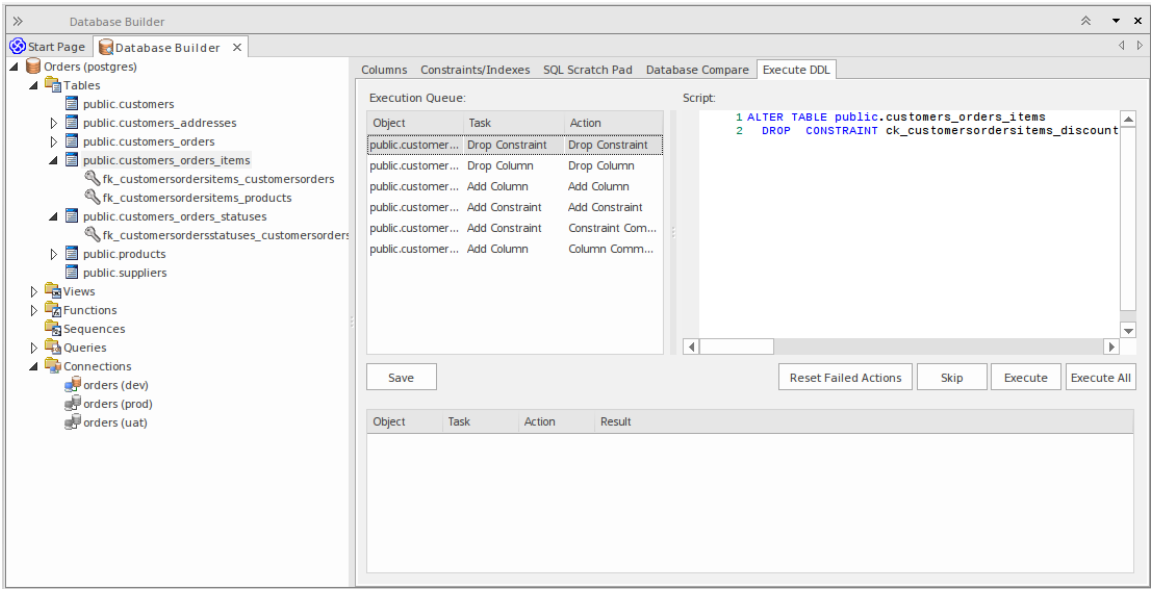


L'onglet « Exécuter DDL » comporte les champs et boutons suivants :

Champ/Bouton	Action
1 file d'attente d'exécution	Répertorie les tâches (chacune avec une instruction DDL associée) qui doivent encore être exécutées. La liste comporte trois colonnes qui précisent le nom de l' object impliqué, la tâche et l'action en cours. La sélection d'un élément dans la liste affichera l'instruction DDL associée (dans le champ 'Script') pour la tâche donnée.
2 Scénario	Une zone de texte avec coloration syntaxique SQL, affichant l'instruction DDL pour la tâche sélectionnée.
3 Enregistrer	Cliquez sur ce bouton pour enregistrer toutes les instructions DDL individuelles de la « file d'attente d'exécution » et de la « liste des résultats » dans un seul fichier.
4 Réinitialiser les actions ayant échoué	Cliquez sur ce bouton pour remettre en file d'attente toutes les tâches ayant échoué ou ignorées de la « Liste des résultats » vers le bas de la « File d'attente d'exécution ».
5 Sauter	Cliquez sur ce bouton pour sauter la tâche suivante dans la « file d'attente d'exécution » et ne pas l'exécuter. La tâche sera déplacée dans la « Liste des résultats » et ne recevra aucun résultat. Lorsque vous cliquez sur le bouton Réinitialiser les actions ayant échoué, les tâches ignorées sont renvoyées dans la file d'attente d'exécution avec toutes les tâches ayant échoué.
6 Exécuter	Cliquez sur ce bouton pour exécuter la tâche suivante dans la « file d'attente d'exécution ». La tâche est supprimée du haut de la « file d'attente d'exécution » et ajoutée à la fin de la « liste des résultats » avec le résultat de l'exécution.
7 Exécuter tout	Cliquez sur ce bouton pour exécuter toutes les tâches dans la « file d'attente d'exécution ». Une fois l'exécution terminée, la « Liste des résultats » affichera les résultats de chaque tâche individuelle.
8 Liste des résultats	Répertorie les tâches exécutées avec les résultats d'exécution pour chaque tâche. La sélection d'un élément dans cette liste affichera l'instruction DDL qui a été exécutée, dans le champ « Script ».

Exemple

Dans l'exemple utilisé dans la section précédente sur la comparaison de bases de données (lorsqu'une colonne et une contrainte ont été renommées), si les valeurs par défaut sont utilisées pour « pousser » les modifications du modèle de données dans la base de données active, l'écran Exécuter DDL est renseigné avec les détails affichés ici.



En résumé, DDL est généré pour supprimer à la fois l'ancienne colonne et l'ancienne contrainte (tâches 'Supprimer la colonne' et 'Supprimer la contrainte'), puis la colonne et la contrainte sont créées avec les nouveaux noms (tâches 'Ajouter une colonne' et 'Ajouter une contrainte'). ') et enfin chacun voit ses commentaires/remarques appliqués (tâches 'Ajouter une contrainte - Commentaire de contrainte' et 'Ajouter une colonne - Commentaire de colonne').

Comparaison de base de données

L'onglet « Comparaison de base de données » fournit un mécanisme permettant de comparer le modèle de données actuel avec une base de données active et, éventuellement, de synchroniser les différences dans les deux sens. Les différences « poussées » dans une base de données active sont effectuées à l'aide des instructions « Alter DDL », tandis que les modifications importées de la base de données active peuvent être directement « extraites » dans le modèle.

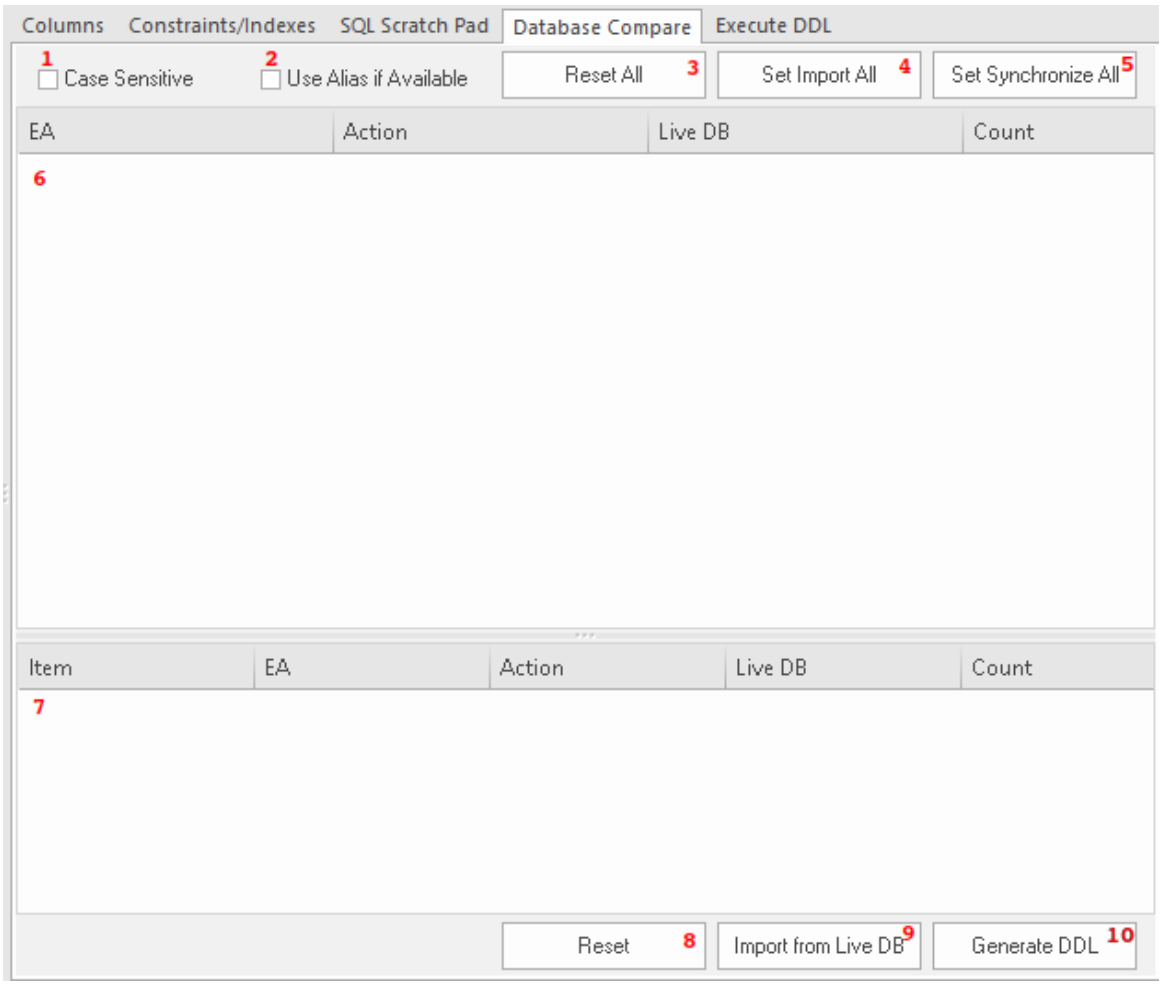
La fonctionnalité Database Compare nécessite que Database Builder dispose d'une connexion valide à une base de données active. Cette connexion à la base de données est partagée par les onglets « SQL Scratch Pad », « Database Compare » et « Execute DDL » du Database Builder.

Accéder

Ouvrez la fenêtre Database Builder, puis affichez l'onglet « Database Compare ».

Ruban	Développement > Modélisation de données > Générateur de base de données > Comparaison de base de données

L'onglet Comparaison DDL



L'onglet « Comparaison de base de données » comporte un certain nombre de contrôles, comme décrit ici.

Numéro et nom	Description
1 sensible à la casse	Cliquez sur cette case à cocher pour que toutes les comparaisons de propriétés reconnaissent les différences de casse dans le texte de la propriété.
2 Utilisez l'alias si disponible	Cochez cette case pour indiquer que tous les alias définis doivent être utilisés à la place des noms object (au niveau object et de la colonne).
3 Réinitialiser tout	Cliquez sur ce bouton pour remettre le drapeau ' Action ' de tous les objets à la valeur par défaut.
4 Définir tout importer	Cliquez sur ce bouton pour définir le flag ' Action ' de toutes les différences détectées à <====; c'est-à-dire mettre à jour le modèle avec la ou les valeur de la base de données en direct.
5 Définir tout synchroniser	Cliquez sur ce bouton pour définir le drapeau ' Action ' de toutes les différences détectées à ====> ; c'est-à-dire mettre à jour la base de données en direct avec la ou valeur du modèle.
6 différences	Révision la liste des objets présentant des discordances entre le modèle et la base de données active. La sélection d'un élément dans cette liste remplira la liste « Composants ». (Voir le tableau *Liste des différences* pour une description détaillée de chaque colonne.)
7 composants	Révision cette liste de propriétés de l' object sélectionné qui diffèrent entre le modèle et la base de données active. (Voir le tableau *Liste des composants* pour une description détaillée de chaque colonne.)
8 Réinitialiser	Cliquez sur ce bouton pour remettre le drapeau ' Action ' pour toutes les propriétés de l' object actuel à la valeur par défaut.
9 Importer depuis Live DB	Cliquez sur ce bouton pour importer les valeurs de toutes les propriétés (avec l'" Action ' de <===) de la base de données en direct dans le modèle.
10 Générer DDL	Cliquez sur ce bouton pour générer les instructions 'Alter DDL' pour tous les objets avec une ' Action ' de ====>, et envoyer les instructions vers l'onglet 'Execute DDL'.

Liste des différences

Colonne	Description
EA	Affiche le nom de chaque object du modèle qui présente une ou plusieurs différences détectées. Les valeurs vides indiquent que l' object est manquant dans le modèle mais existe dans la base de données active.
Action	La valeur par défaut est « Aucune Action » comme action à entreprendre en tenant compte de la ou des différences de cet object . Cliquez sur la flèche déroulante et sélectionnez une action spécifique. La liste des actions disponibles dans la liste

	dépendra du fait que l' object donné soit associé ou non dans le modèle et la base de données active. **Objets appariés** • Aucune Action : ne mettez pas à jour la base de données ou le modèle avec cette modification. • ===> - mettre à jour l' object dans la base de données à partir du modèle • <=== - mettre à jour l' object dans le modèle à partir de la base de données • Personnaliser : définissez les éléments sur Aucune Action avant de définir différentes actions sur chaque élément dans le panneau inférieur. • Dissocier - séparez les objets appariés afin qu'ils ne soient pas comparés les uns aux autres ou mis à jour les uns par rapport aux autres **Objets non appariés** • Créer <nom object > - crée l' object de base de données manquant dans la base de données ou le modèle, selon le cas • Supprimer <nom object > - supprime l' object du modèle • Drop <nom object > - supprime l' object de la base de données • Associer avec <nom object > - associez l' object dans la base de données à l' object nommé (non apparié) dans le modèle, afin qu'ils soient comparés pour les différences entre eux Les champs « Action » dans la « Liste des composants » (le panneau inférieur) seront mis à jour en fonction de la sélection de ce champ. Par exemple, si la base de données active a une colonne Tableau « Adresse1 » et que le modèle n'en a pas, définir l' object « Action » sur « **===> »** (mettre à jour l' object dans la base de données à partir du modèle) définit la colonne « Item ». Action ' vers ' Supprimer l'adresse 1', ce qui supprimera la colonne de la base de données en direct.
Base de données en direct	Affiche le nom de chaque object de la base de données active qui présente une ou plusieurs différences détectées. Les valeurs vides indiquent que l' object existe dans le modèle mais est manquant dans la base de données active.
Compter	Affiche le nombre total de différences détectées pour l' object (et tous ses composants) entre le modèle et la base de données active.

Liste des composants

Colonne	Description
Item	Affiche le nom ou la description du composant pour chaque différence détectée. Les différences sont regroupées en trois catégories : Propriétés , Colonnes et Contraintes, dans une arborescence.
EA	Affiche la valeur du composant donné tel que détecté dans le modèle. Les valeurs vides indiquent que la valeur est manquante dans le modèle mais existe dans la base de données active.
Action	Par défaut, l'action correspondant au paramétrage du champ ' Action ' de l' object dans la liste 'Différences', pour indiquer l'action à entreprendre concernant la

	différence détectée pour le composant. Cliquez sur la flèche déroulante pour sélectionner une action alternative ; les options disponibles dans la liste dépendent du type du composant et de la différence détectée. • Aucune Action - ne pas mettre à jour la base de données ou le modèle • ===> - mettre à jour l' object dans la base de données en direct à partir du modèle • <=== - mettre à jour l' object dans le modèle à partir de la base de données en direct • Ajouter <nom de l'élément> - créez l'élément manquant dans la base de données ou le modèle, selon le cas • Supprimer <nom de l'élément> - supprime l'élément du modèle • Drop <item name> - supprime l'élément de la base de données active
Base de données en direct	Affiche la valeur du composant sélectionné dans la base de données en direct. Les valeurs vides indiquent que la valeur existe dans le modèle mais est manquante dans la base de données active.
Compter	Affiche le nombre de différences entre le modèle et la base de données active détectées dans le composant sélectionné.

Travailler avec la comparaison de bases de données

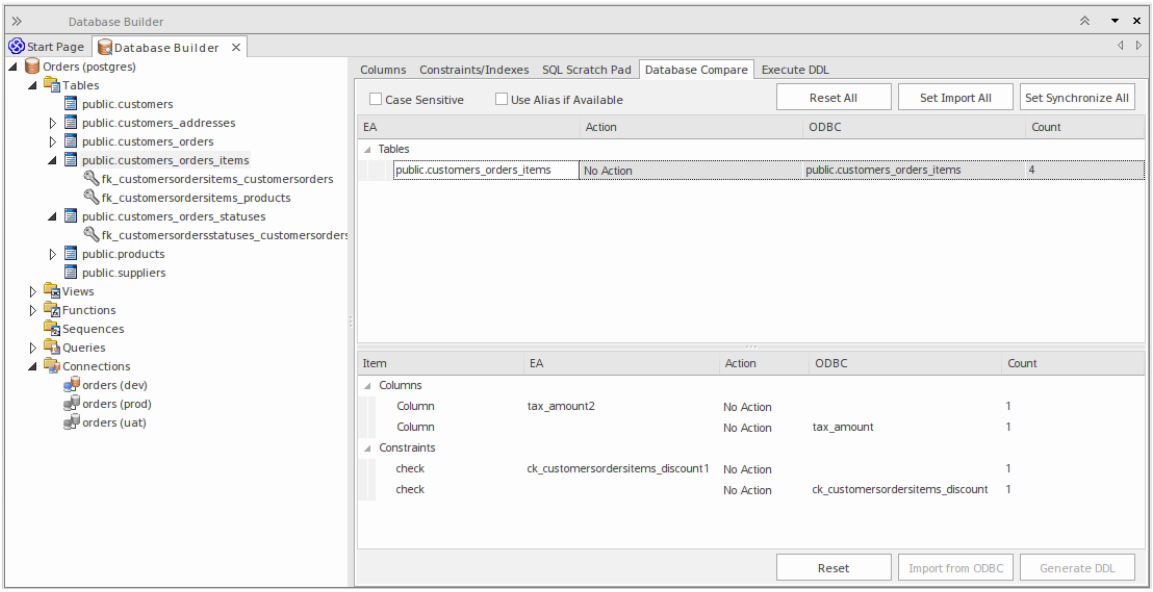
Chaque fois que vous effectuez une comparaison, Enterprise Architect lit la définition à la fois de la base de données active et du modèle, puis tente de « coupler » chaque object d'une source avec l'autre, en utilisant son nom (et son schéma, si cela est pertinent pour le SGBD actuel). .

Si une correspondance est trouvée, le nom object est affiché dans les colonnes « EA » et « Live DB » avec une action par défaut de « Aucune Action ». La colonne « Nombre » indique le nombre total de différences trouvées pour l' object et ses composants ou propriétés.

Si aucune correspondance n'est trouvée entre les systèmes, le nom object est affiché dans la colonne source (soit « EA » soit « Live DB ») tandis que l'autre colonne est vide. Dans cet état, il est possible d'associer l' object à un object d'un nom différent ; la liste déroulante ' Action ' présentera les objets disponibles. Si un nouveau couplage est effectué, les définitions des deux objets sont comparées pour détecter les différences et les résultats sont affichés dans la liste « Composants », avec l'action par défaut « ====> » sélectionnée.

Si vous sélectionnez une action au niveau de l' object , cela définira l'action correspondante pour tous les composants et propriétés de l' object . Cependant, si vous sélectionnez l'action « Personnaliser » au niveau de l' object , vous pouvez déterminer une action différente pour chaque composant.

À titre d'exemple, une colonne (tax_amount) et une contrainte (ck_customersordersitems_discount) ont été renommées dans Tableau « public.customers_order_items » (dans l'exemple de modèle) et une comparaison de base de données a été effectuée ; cette image montre les différences trouvées :



Dans l'image, un seul Tableau a détecté des différences : « public.customers_order_items » ; en sélectionnant cette option, la liste « Composants » est remplie. À partir des résultats détectés, il peut être déterminé que le modèle de données contient une colonne (tax_amount2) et une contrainte de contrôle (ck_customerordersitems_discount1) que la base de données en direct ne contient pas et, à son tour, la base de données en direct contient une colonne (tax_amount) et une contrainte de contrôle (ck_customerordersitems_discount) que le modèle de données ne fait pas.

Comparaison avec les options

La fonctionnalité « Comparer avec les options » fonctionne de la même manière que pour une comparaison directe, sauf que vous êtes invité à choisir quelles comparaisons object /propriétés doivent être effectuées. Cela vous permet d'ignorer certaines différences qui ne sont pas pertinentes à l'heure actuelle.

Ces tableaux décrivent les différentes comparaisons qui peuvent être activées ou désactivées.

Tous les objets, propriétaire

Comparaison	Action
Propriétaire	Sélectionnez cette option pour indiquer que la propriété « Propriétaire » de tous les objets de base de données doit être comparée, une fois que les objets ont été « couplés ».

Options Tableau

Option	Action
Tableaux	Sélectionnez cette option parent pour activer toutes les options de comparaison Tableau . Désélectionnez pour désactiver toutes les autres options. Vous

	désélectionnerez ou sélectionnerez ensuite des options spécifiques dans la liste.
Tableau - Propriétés étendues	Sélectionnez cette option pour indiquer que les propriétés étendues des Tableaux (telles que la version de base de données et l'espace de table) doivent être comparées.
Tableau - Remarques	Sélectionnez pour indiquer que les remarques appliquées aux Tableaux doivent être comparées.
Colonnes	Sélectionnez cette option parent pour activer toutes les options de « Comparaison de colonnes ». Désélectionnez pour désactiver toutes les autres options « Colonne ». Vous désélectionnerez ou sélectionnerez ensuite des options spécifiques dans la liste.
Colonne - Type	Sélectionnez cette option pour indiquer que le nom du type de données des colonnes Tableau doit être comparé.
Colonne - Taille	Sélectionnez cette option pour indiquer que la taille du type de données des colonnes Tableau doit être comparée.
Colonne - Valeur par défaut	Sélectionnez pour indiquer que les valeurs par défaut des colonnes Tableau doivent être comparées.
Colonne - Position	Sélectionnez pour indiquer que les positions des colonnes Tableau doivent être comparées.
Colonne - Non nulle	Sélectionnez pour indiquer que la propriété non nulle des colonnes Tableau doit être comparée.
Colonne - Numérotation automatique	Sélectionnez cette option pour indiquer que les propriétés de numérotation automatique des colonnes Tableau doivent être comparées (telles que AutoNum, StartNum et Increment).
Colonne - Colonnes sans correspondance	Sélectionnez cette option pour indiquer que les colonnes Tableau qui ne correspondent pas entre le modèle et la base de données active doivent être comparées. Il s'agit généralement de colonnes qui existent dans un système mais qui n'existent pas dans l'autre.
Colonne - Propriétés étendues	Sélectionnez cette option pour indiquer que les propriétés étendues des colonnes Tableau (telles que Unsigned et Zerofill) doivent être comparées.
Colonne - Remarques	Sélectionnez cette option pour indiquer que les remarques appliquées aux colonnes Tableau doivent être comparées.
Contraintes	Sélectionnez cette option parent pour activer toutes les options de « Comparaison des contraintes Tableau ». Désélectionnez pour désactiver toutes les options « Contrainte Tableau ». Vous désélectionnerez ou sélectionnerez ensuite des options spécifiques dans la liste.
Contrainte - Primary Keys	Sélectionnez cette option pour indiquer que les propriétés liées aux Primary Keys doivent être comparées.
Contrainte - Foreign Keys	Sélectionnez cette option pour indiquer que les propriétés liées aux Foreign Keys doivent être comparées.

Contrainte - Index	Sélectionnez cette option pour indiquer que les propriétés liées aux index doivent être comparées.
Contrainte - Contraintes uniques	Sélectionnez cette option pour indiquer que les propriétés liées aux contraintes uniques doivent être comparées.
Contrainte - Vérifier les contraintes	Sélectionnez cette option pour indiquer que les propriétés liées aux contraintes de vérification doivent être comparées.
Contrainte - Tableau Déclencheurs	Sélectionnez cette option pour indiquer que les propriétés liées à Tableau Déclencheurs doivent être comparées.
Contrainte - Contraintes inégalées	Sélectionnez cette option pour indiquer que les contraintes Tableau qui ne correspondent pas entre le modèle et la base de données active doivent être comparées. Il s'agit généralement de contraintes qui existent dans un système mais qui n'existent pas dans l'autre.
Contraintes - Propriétés étendues	Sélectionnez cette option pour indiquer que les propriétés étendues des contraintes Tableau (telles que Fill Factor et Clustered) doivent être comparées.
Contraintes - Remarques	Sélectionnez cette option pour indiquer que les remarques appliquées aux contraintes Tableau doivent être comparées.

Notes

- La fonctionnalité Database Compare peut actuellement effectuer des comparaisons sur les types object Tableau , Vue , Procedure, Function et Séquence .

Créer un Diagramme Modèle de données

Pour modéliser la structure d'une base de données relationnelle, vous utilisez diagrammes Modélisation de données, qui sont diagrammes de classes étendus. Lorsque vous ouvrez un diagramme Modélisation de données, la boîte à outils Diagramme correspondante s'ouvre automatiquement et contient les éléments diagramme :

- Tableau
- Vue
- Procédure
- Séquence
- Fonction
- Association et
- Connexion à la base de données

Accéder

Affichez la dialogue ' Nouveau Diagramme ' en utilisant l'une des méthodes décrites ici.

Ruban	Conception > Diagramme > Ajouter Diagramme
Menu Contexte	Cliquez-droit sur Paquetage | Ajouter Diagramme Cliquez-droit sur l'élément | Ajouter | Ajouter Diagramme
Raccourcis Clavier	Ctrl+Inser
Autre	Menu de la barre de légende de la fenêtre Navigateur | Nouveau Diagramme

Créer un diagramme Modélisation de données

Champ/Bouton	Action
Paquetage	La valeur par défaut est le nom du Paquetage sélectionné dans la fenêtre Navigateur ou, si le parent est un élément, le nom du Paquetage contenant cet élément. Si vous ajoutez un diagramme directement à un Paquetage et remarquez qu'il ne s'agit pas du bon Paquetage , cliquez sur le bouton et recherchez le bon Paquetage .
Parent	Si vous ajoutez un diagramme à un élément, ce champ affiche le nom de l'élément.
Diagramme	Ce champ indique par défaut le nom du Paquetage ou de l'élément parent. Si nécessaire, remplacez le nom par défaut par votre nom préféré.
Sélectionner à partir de	Cliquez sur cet en-tête et sélectionnez le groupe de perspectives et la perspective ou l'espace de travail les plus appropriés au domaine dans lequel vous travaillez (dans ce cas, « Ingénierie de l'information > Modèles de base de données »).

	Parmi les options répertoriées dans le panneau, cliquez sur « Étendu ».
Types Diagramme	Cliquez sur ' Modélisation des données'.
OK	Cliquez sur ce bouton pour créer le diagramme . Le Diagramme Vue affiche le diagramme vierge, et les pages ' Modélisation des données' s'affichent dans la Boîte à outils Diagramme . Faites glisser les éléments et les connecteurs de la boîte à outils sur votre diagramme pour créer votre modèle de données.

Notes

- La notation par défaut du connecteur diagramme pour tous les nouveaux diagrammes est Information Engineering, bien que de nombreux modélisateurs de données préfèrent la notation IDEF1X ; pour effectuer cette modification, sélectionnez « Conception > Diagramme > Gérer > Propriétés > Connecteurs » et cliquez sur l'option requise dans la liste déroulante « Notation du connecteur ».

Exemple Diagramme Modèle données

Cet exemple de diagramme Data Modèle montre un modèle de données d'un système d'entreposage de librairie. Les tableaux sont modélisés à l'aide d'une classe stéréotypée avec un compartiment pour les colonnes qui affiche le nom et le type des colonnes. Foreign Keys sont indiquées par des stéréotypes sur les colonnes. Vous pouvez examiner ce modèle plus en détail dans l'exemple de modèle, installé avec Enterprise Architect et disponible à partir de cet emplacement du ruban.

Démarrer > Aide > Aide > Ouvrir l'exemple Modèle

diagramme modélisation de données avec compartiment d'opération supprimé montrant tableaux connectés pour indiquer les relations foreign key .

Créer un Modèle de Données à partir d'un Modèle Motif

Le moyen le plus simple de créer un espace de travail Modélisation de données est d'utiliser la base de données Modèle Motifs prédéfinie, disponible via l' Assistant de Modèle (onglet Page Démarrage 'Créer à partir de Motif '). Enterprise Architect fournit un Motif pour chaque SGBD pris en charge par le système.

Accéder

Affichez l' Assistant de Modèle (onglet Page Démarrage 'Créer à partir Motif ') en utilisant l'une des méthodes décrites ici.

Dans l' Assistant de Modèle , sélectionnez la Perspective ' Ingénierie Base de Données '.

Ruban	Design > Paquetage > Assistant de Modèle
Menu Contexte	Cliquez-droit sur Paquetage | Ajouter un Modèle à l'aide de Assistant
Raccourcis Clavier	Ctrl+Maj+M
Autre	Menu de la barre de légende de la fenêtre Navigateur | Nouveau Modèle de Motif

Créer un Modèle de données

Champ/Bouton	Action
Ajouter au Paquetage	Affiche le nom du Paquetage racine sélectionné.
Technologie	Cliquez sur 'Base de données'.
Nom	Si nécessaire, développez le groupe Ingénierie Base de Données de Motifs . Cochez la case en regard de chaque système de gestion de base de données que vous prenez en charge dans le modèle.
Tous	Cliquez sur ce bouton pour cocher les cases de tous les types de modèles Ingénierie Base de Données et du diagramme Entité-Relation, afin de les inclure tous dans le modèle.
Aucun	Cliquez sur ce bouton pour décocher toutes les cases sélectionnées afin de pouvoir re-sélectionner certaines cases individuellement.
OK	Cliquez sur ce bouton pour ajouter à la fenêtre Navigateur les Paquetages et diagramme pour chaque système de gestion de base de données que vous modélisation .

Ce que chaque Motif Modélisation de données apporte

- Un diagramme récapitulatif du modèle
- Un élément Artefact Rapport Spécification (sur le diagramme récapitulatif) qui peut être utilisé pour documenter rapidement le modèle de données
- Un Paquetage pour chacun des modèles logiques et physiques
- Au sein du Modèle Physique Paquetage , une hiérarchie prédéfinie de sous-Packages, un pour chaque type object pris en charge par le SGBD modélisé (tels que Tableaux , Vues , Procédures et Fonctions) ; ceux-ci organisent automatiquement les objets de la base de données au fur et à mesure de leur ajout
- Le type de SGBD pour l'espace de travail
- Un propriétaire par défaut
- Un diagramme Modélisation de données dans chaque Paquetage avec la notation du connecteur définie sur IDEF1X

Notes

- Une fois qu'un espace de travail modélisation de données a été créé, vous pouvez commencer à développer votre modèle de deux manières :
  - Via Database Builder, une vue spécialement conçue pour supporte les modélisateurs de bases de données.
  - Via la fenêtre Navigateur et diagrammes , qui est la méthode traditionnelle qui pourrait convenir aux utilisateurs qui sont des modélisateurs UML expérimentés

Démarrage

Les modélisateurs d'informations, les modélisateurs de données et les architectes sont chargés de créer des modèles d'informations d'une organisation qui couvrent plusieurs niveaux d'abstraction, du conceptuel au logique et physique. Les modèles conceptuels sont indépendants de la technologie et peuvent être utilisés pour des discussions avec des hommes d'affaires et des experts du domaine, permettant aux concepts de base du domaine d'être représentés, discutés et convenus. Le modèle logique élabore le modèle conceptuel, en ajoutant plus de détails et de précision, mais reste généralement neutre sur le plan technologique, permettant aux analystes de l'information de discuter et de se mettre d'accord sur les structures logiques. Le modèle physique applique des données technologiques spécifiques aux modèles et permet aux ingénieurs de discuter et de se mettre d'accord sur des décisions technologiques en vue de la génération dans un environnement cible, tel qu'un système de gestion de base de données.

Sélection de la perspective

Enterprise Architect partitionne les fonctionnalités étendues de l'outil dans Perspectives , ce qui garantit que vous pouvez vous concentrer sur une tâche spécifique et travailler avec les outils dont vous avez besoin sans être distrait par d'autres fonctionnalités . Pour travailler avec les fonctionnalités Modélisation de Données vous devez d'abord sélectionner l'une de ces Perspectives :

<nom de la perspective> > Ingénierie Base de Données > Ingénierie Base de Données

<nom de la perspective> > Ingénierie Base de Données > Relations Entité

La configuration de la Perspective garantit que les diagrammes Ingénierie Base de Données , leurs boîtes à outils et autres fonctionnalités de la Perspective seront disponibles par défaut.

Exemple Diagramme

Un exemple diagramme fournit une introduction visuelle au sujet et vous permet de voir certains des éléments et connecteurs importants créés lors de la spécification ou de la description de la manière dont un modèle de données est défini, notamment : Tableaux , Vues , Procédures, Séquences, Fonctions.

Types Modèle de données

Les informations peuvent être modélisées à plusieurs niveaux d'abstraction, en commençant par un modèle conceptuel généralement créé par ou pour les hommes d'affaires, un modèle logique utilisé par les analystes commerciaux et systèmes et un modèle physique qui relève des technologues tels que ingénieurs de bases de données. Dans cette rubrique, vous apprendrez à gérer les trois niveaux de modèles d'information.

Création et gestion de modèles de données

Dans cette rubrique, vous apprendrez comment travailler en détail à l'aide Enterprise Architect pour gérer votre schéma de base de données physique. Cela inclut l'utilisation de l'outil Database Builder qui vous permet d'interagir avec n'importe quel nombre de bases de données en direct via une connexion ODBC.

Importer le schéma de base de données

Cette rubrique vous montrera comment vous connecter à une base de données en direct, y compris les systèmes de production, Test et de développement, et procéder à l'ingénierie inverse de la base de données dans un modèle créant Tableaux , Vues , des procédures, une intégrité référentielle déclarative, etc. Un diagramme de la base de données est automatiquement créé et les éléments tels que tableaux peuvent être liés à d'autres éléments du modèle, notamment des modèles conceptuels et logiques, des tests de cours de programmation, etc.

Générer un langage de définition de base de données (DDL)

Dans cette rubrique, vous apprendrez comment exploiter la puissance des modèles de données en générant du code Database Definition Language directement à partir du modèle. Enterprise Architect peut générer du code dans une large gamme de systèmes de gestion de bases de données.

Systèmes de gestion de base de données pris en charge

Enterprise Architect offre support riche pour la plupart des principaux systèmes de gestion de bases de données (SGBD). Cette fonctionnalité permet de comparer des modèles de systèmes disparates soit pour la génération de code, soit pour l'analyse en utilisant la fonctionnalité d'importation. Cette rubrique répertorie les SGBD pris en charge et

Plus d'information

Cette section fournit des liens utiles vers d'autres sujets et ressources qui pourraient vous être utiles lorsque vous travaillez avec les fonctionnalités de l'outil Modélisation de données.