

IBM Cognos Framework Manager  
Version 10.2.2

*Guide de modélisation des  
métadonnées*



**Important**

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques», à la page 55.

Certaines illustrations de ce manuel ne sont pas disponibles en français à la date d'édition.

LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFAÇON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.ibm.com/ca/fr> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France  
Direction Qualité  
17, avenue de l'Europe  
92275 Bois-Colombes Cedex*

Le présent document s'applique à IBM Cognos Business Intelligence version 10.2.2 et peut également s'appliquer aux éditions suivantes.

© Copyright IBM Corporation 2005, 2014.

---

# Table des matières

<b>Avis aux lecteurs canadiens</b> . . . . .	<b>v</b>
<b>Introduction</b> . . . . .	<b>vii</b>
<b>Chapitre 1. Guide de modélisation des métadonnées</b> . . . . .	<b>1</b>
Présentation des concepts de modélisation d'IBM Cognos . . . . .	1
Concepts de modélisation relationnelle . . . . .	1
Considérations de conception de modèles . . . . .	11
Concepts de modélisation dimensionnelle . . . . .	19
Construction du modèle relationnel . . . . .	21
Définition de la base de la modélisation relationnelle . . . . .	22
Définition de la représentation dimensionnelle du modèle . . . . .	31
Organisation du modèle . . . . .	34
<b>Chapitre 2. Code SQL généré par IBM Cognos Software</b> . . . . .	<b>37</b>
Présentation des requêtes dimensionnelles . . . . .	37
Requête à fait unique . . . . .	37
Requête à granularité multiple et à faits multiples portant sur des dimensions conformes . . . . .	39
Modélisation de relations de type 1-n en relations de type 1-1 . . . . .	42
Requête à granularité multiple et à faits multiples portant sur des dimensions non conformes . . . . .	43
Résolutions des dimensions et faits identifiés de façon ambiguë . . . . .	48
Sujets de requête représentant un niveau de hiérarchie . . . . .	48
Résolution de requêtes fractionnées par erreur . . . . .	49
<b>Remarques</b> . . . . .	<b>55</b>
<b>Index</b> . . . . .	<b>59</b>



---

## Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

### Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

### Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

### Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.








### OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

### Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
 (Pos1)		Home
Fin	Fin	End
 (PgAr)		PgUp
 (PgAv)		PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
 (Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

## Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

## Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

---

## Introduction

IBM® Cognos Framework Manager est un outil de modélisation des données. Un modèle est une présentation professionnelle des informations contenues dans une ou plusieurs sources de données. Lorsque vous ajoutez des fonctions de sécurité et des fonctions multilingues à cette présentation professionnelle, un modèle unique est capable de répondre aux besoins de nombreux groupes d'utilisateurs à travers le monde.

Le présent document traite des concepts fondamentaux de la modélisation des métadonnées dans IBM Cognos en vue de leur utilisation dans le cadre de la génération de rapports professionnels et de l'analyse de données. Il décrit également la construction du modèle relationnel.

### Audience

Ce document est destiné à permettre la compréhension des concepts de modélisation d'IBM Cognos.

### Recherche d'informations

Pour rechercher la documentation des produits sur le Web, y compris toutes les documentations traduites, accédez au centre de documentation IBM Knowledge Center (<http://www.ibm.com/support/knowledgecenter>).

### Instructions prospectives

La présente documentation décrit les fonctionnalités actuelles du produit. Elle peut contenir des références à des éléments qui ne sont pas disponibles actuellement. Cela n'implique aucune disponibilité ultérieure de ces éléments. De telles références ne constituent en aucun cas un engagement, une promesse ou une obligation légale de fournir un élément, un code ou une fonctionnalité. Le développement, la disponibilité et le calendrier de mise à disposition des fonctions demeurent à la seule discrétion d'IBM.

### Clause de décharge relative aux exemples

La société Vacances et Aventure, Ventes VA, ou toute variation du nom Vacances et Aventure, ainsi que les exemples de planification, illustrent des opérations commerciales fictives, avec des exemples de données utilisées pour développer des exemples d'applications, destinées à IBM et à ses clients. Les exemples de données contenus dans ces enregistrements fictifs sont utilisés pour représenter des transactions de vente, la distribution de produits, des données financières et des ressources humaines. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite. D'autres fichiers d'exemple peuvent contenir des données fictives générées manuellement ou par une machine, des données factuelles compilées à partir de sources académiques ou publiques, ou des données utilisées avec l'autorisation du détenteur des droits d'auteur, à utiliser comme exemple de données pour développer des exemples d'application. Les noms de produit référencés peuvent être les marques de leurs propriétaires respectifs. Toute reproduction sans autorisation est interdite.

## Fonctions d'accessibilité

Les fonctions d'accessibilité permettent aux utilisateurs souffrant d'un handicap physique, comme une mobilité réduite ou une vision limitée, d'utiliser les produits informatiques. IBM Cognos Framework Manager est doté de fonctions d'accessibilité. Pour en savoir davantage sur ces fonctionnalités, reportez-vous au manuel Framework Manager *User Guide*.



---

## Chapitre 1. Guide de modélisation des métadonnées

IBM Cognos Framework Manager est un outil de modélisation des données qui permet de générer des requêtes pour IBM Cognos BI. Un modèle est un ensemble de métadonnées comprenant des informations physiques et professionnelles pour une ou plusieurs sources de données. IBM Cognos BI permet la gestion des performances sur des données relationnelles normalisées ou dénormalisées, ainsi que sur diverses sources de données OLAP.

Pour ouvrir le *Guide de modélisation des métadonnées IBM Cognos* dans une autre langue, accédez à *emplacement\_installation\c10\webcontent\documentation* et ouvrez le dossier pour la langue souhaitée. Puis ouvrez le fichier *ug\_best.pdf*.

---

### Présentation des concepts de modélisation d'IBM Cognos

Avant de commencer vous devez comprendre les concepts fondamentaux d'IBM Cognos en matière de modélisation des métadonnées à utiliser dans la génération de rapports professionnels et l'analyse de données.

#### Concepts de modélisation relationnelle

Lors d'une modélisation dans IBM Cognos Framework Manager, il est important de comprendre que vous n'êtes pas obligé de concevoir votre source de données selon un schéma en étoile parfait. Les schémas en flocon de neige et autres formes de schémas normalisés sont tout aussi acceptables si votre source de données est optimisée pour fournir les performances requises pour l'application. Il est généralement recommandé de créer un modèle logique correspondant aux concepts du schéma en étoile. Cette condition, requise pour IBM Cognos Analysis Studio a par ailleurs déjà fait ses preuves en tant que moyen d'organisation efficace des données pour vos utilisateurs.

Lorsque vous commencez à développer votre application avec une source de données complexe, il est conseillé de créer une vue simplifiée représentant la façon dont vos utilisateurs perçoivent l'activité, en la concevant à l'aide des consignes fournies dans le présent document, afin d'obtenir les requêtes et résultats attendus. Un modèle relationnel bien construit sert de base à votre application et vous fournit un point de départ fiable pour tirer parti des fonctionnalités dimensionnelles des logiciels IBM Cognos.

Si vous commencez avec une source de données à schéma en étoile, le modèle est plus simple à réaliser car les concepts utilisés dans la création d'un schéma en étoile se prêtent bien à la construction d'applications de requête et d'analyse. Les consignes fournies dans le présent document vous aideront à concevoir un modèle répondant aux besoins de votre application.

#### Cardinalité

Des relations existent entre deux sujets de requête. La cardinalité d'une relation est le nombre de lignes liées à chacun des deux sujets de requête. Les lignes sont liées par l'expression de la relation ; cette expression fait généralement référence aux clés primaires et étrangères des tables sous-jacentes.

Le logiciel IBM Cognos utilise la cardinalité d'une relation pour :

- éviter le double comptage des données de fait,
- prendre en charge les jointures de boucles qui sont fréquentes dans les modèles à schéma en étoile,
- optimiser l'accès au système de la source de données sous-jacente,
- identifier les sujets de requête qui se comportent comme des faits ou dimensions.

Une requête qui utilise plusieurs faits de différentes tables sous-jacentes est divisée en requêtes distinctes pour chaque table de faits sous-jacente. Chaque requête de fait individuelle fait référence à sa table de faits correspondante ainsi qu'aux tables dimensionnelles liées à cette table de faits. Une autre requête est utilisée pour fusionner ces requêtes individuelles en ensemble unique de résultats. La requête de cette dernière opération est généralement appelée requête liée. Vous savez qu'il s'agit d'une requête liée lorsque vous pouvez voir coalesce ainsi qu'une jointure externe intégrale.

La requête liée permet également au logiciel IBM Cognos de mettre correctement des données en rapport à divers niveaux de granularité. Voir «Requêtes à faits multiples, requêtes à granularité multiple», à la page 8.

#### **Cardinalité des requêtes générées :**

Le logiciel IBM Cognos prend en charge la cardinalité minimale/maximale et la cardinalité facultative.

Dans  $0:1$ ,  $0$  est la cardinalité minimale et  $1$  la cardinalité maximale.

Dans  $1:n$ ,  $1$  est la cardinalité minimale et  $n$  la cardinalité maximale.

Une relation dont la cardinalité définie est  $1:1$  à  $1:n$  est généralement considérée comme  $1$  à  $n$  lors de l'analyse des cardinalités maximales.

Une cardinalité minimale de  $0$  signifie que la relation est facultative. Vous pouvez définir une cardinalité minimale de  $0$  si vous voulez que la requête conserve les informations de l'autre côté de la relation en l'absence de correspondance. Par exemple, une relation entre le client et les ventes réelles peut être définie comme  $1:1$  à  $0:n$ . Ceci indique que les rapports vont afficher les informations relatives au client demandé même s'il n'y a aucune donnée sur les ventes.

Ainsi, une relation  $1$  à  $n$  peut également être indiquée sous la forme :

- $0:1$  à  $0:n$
- $0:1$  à  $1:n$
- $1:1$  à  $0:n$
- $1:1$  à  $1:n$

Utilisez l'instruction **Relationship impact** dans la boîte de dialogue **Relationship Definition** pour mieux comprendre la cardinalité. Par exemple, Sales Staff ( $1:1$ ) est joint à Orders ( $0:n$ ).

Relationship impact:	Each Order has one and only one Sales Staff. Each Sales Staff has zero or more Order (outer join).
----------------------	---

Il est important de vérifier que la cardinalité est correctement prise en compte dans le modèle, car elle détermine la détection des sujets de requête de faits et permet d'éviter le double comptage des données factuelles.

Le logiciel IBM Cognos utilise les règles de base suivantes pour appliquer la cardinalité lors de la génération des requêtes :

- La cardinalité est appliquée dans le contexte d'une requête.
- La cardinalité 1 à n implique des données de fait du côté n et des données de dimension du côté 1.
- Un sujet de requête peut se comporter en tant que sujet de requête de fait ou en tant que sujet de requête dimensionnel, selon les relations requises pour répondre à une requête particulière.

Utilisez **Model Advisor** pour afficher l'évaluation du comportement impliqué par la cardinalité dans votre modèle.

Pour en savoir davantage, reportez-vous aux sections «Requête à fait unique», à la page 37 et «Requête à granularité multiple et à faits multiples portant sur des dimensions conformes», à la page 39.

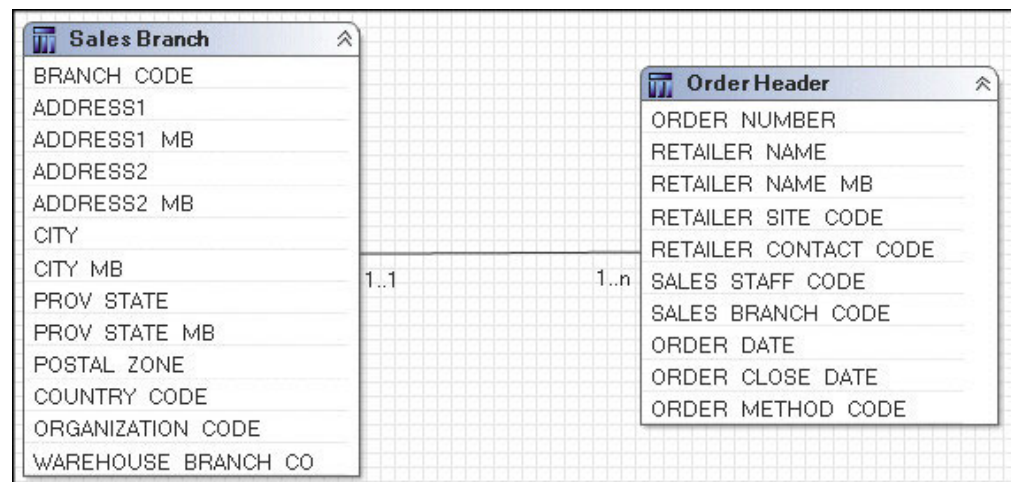
### Cardinalité dans le contexte d'une requête :

Le rôle de la cardinalité dans le contexte d'une requête est essentiel car la cardinalité sert à déterminer où et quand diviser la requête lors de la génération de requêtes à faits multiples. Si les dimensions et les faits ne sont pas bien identifiés, des requêtes liées risquent d'être inutilement créées, ce qui pèse sur les performances, ou bien les requêtes peuvent être formées de façon incorrecte, ce qui peut produire des résultats erronés.

Les exemples ci-dessous illustrent la façon dont la cardinalité est interprétée par le logiciel IBM Cognos.

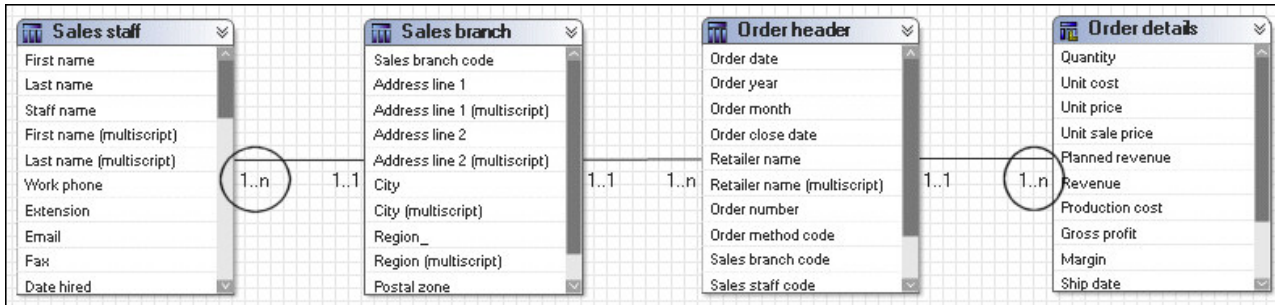
*Exemple : Sujets de requête qui se comportent comme une dimension et un fait :*

Dans cet exemple, le sujet Sales Branch se comporte comme une dimension par rapport au sujet Order Header et le sujet Order Header se comporte comme un fait par rapport au sujet Sales Branch.



Exemple : Quatre sujets de requête inclus dans une requête :

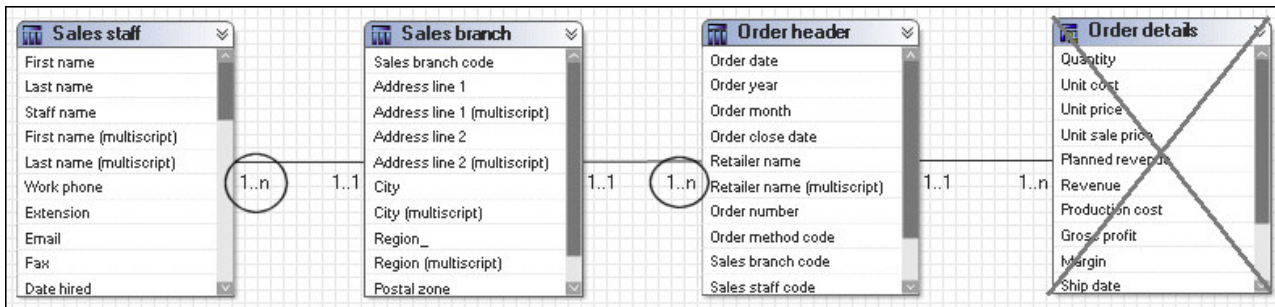
Dans l'exemple qui suit, les quatre sujets de requête sont inclus dans une requête. Sales Staff et Order details sont traités comme des faits. Order header et Sales branch sont traités comme des dimensions.



Le code SQL généré pour cette requête sera divisé, il traitera Sales staff et Order details comme des faits. Les résultats de ces deux sous-requêtes sont liés à l'aide des informations extraites du sujet Sales branch. Cela donne un rapport répertoriant les informations de l'équipe commerciale par succursale à côté des informations sur les commandes et des informations sur les en-têtes de commande par succursale.

Exemple : Trois sujets de requête inclus dans une requête :

Dans l'exemple qui suit, seulement trois sujets de requête sont inclus dans une requête. Le sujet Order details n'est pas utilisé. Order header est désormais traité comme un fait. Sales staff est toujours traité comme un fait.



Le code SQL de cet exemple a également généré une requête liée, qui retourne un résultat similaire à celui obtenu plus haut. Notez qu'une opération liée conserve les informations des deux côtés de l'opération à l'aide d'une jointure externe intégrale.

## Déterminants

Les déterminants reflètent la granularité en représentant des sous-ensembles ou groupes de données dans un sujet de requête, ils servent à garantir l'agrégation correcte de ces données répétées. Les déterminants sont étroitement liés au concept de clés et d'index de la source de données et sont importés en fonction des informations de clé et d'index uniques contenues dans cette source de données. Il est conseillé de toujours vérifier les déterminants importés et, si nécessaire, de les modifier ou d'en créer d'autres. En modifiant les déterminants, vous pouvez

remplacer les informations relatives à l'index et à la clé dans votre source de données par des informations correspondant mieux à vos besoins d'analyse et de génération de rapports. En ajoutant des déterminants, vous pouvez représenter des groupes de données répétées nécessaires à votre application.

L'exemple de la dimension Time ci-dessous représente un exemple de déterminant unique Day. Month est un exemple de déterminant non unique ; dans Month, la clé est répétée selon le nombre de jours d'un mois spécifique. Lorsque vous définissez un déterminant non unique, vous devez en définir le **Regroupement**. Celui-ci indique au logiciel IBM Cognos que, lorsque les clés ou attributs associés à ce déterminant sont répétés dans les données, il convient d'appliquer des fonctions d'agrégat et un groupement afin d'éviter le double comptage. Il est déconseillé de définir des déterminants en sélectionnant à la fois **Identification de façon unique** et **Regroupement**, ou ni l'un ni l'autre.

Clé Year	Clé Month	Nom du mois	Clé Day	Nom du jour
2006	200601	Janvier 06	20060101	Dimanche, Janvier 1, 2006
2006	200601	Janvier 06	20060102	Lundi, Janvier 2, 2006

Vous pouvez définir trois déterminants pour cet ensemble de données de la façon suivante : deux déterminants **Regroupement** (Year et Month) et un déterminant unique (Day). Ce concept est similaire (mais pas identique) au concept de niveaux et de hiérarchies.

Nom du déterminant	Clé	Attributs	Identification de façon unique	Regroupement
Année	Clé Year	Aucun	Non	Oui
Mois	Clé Month	Nom du mois	Non	Oui
Jour	Clé Day	Nom du jour Clé Month Nom du mois Clé Year	Oui	Non

Dans ce cas, une seule clé est utilisée pour chaque déterminant car celle-ci contient suffisamment d'informations pour identifier un groupe parmi les données. Le sujet Month est parfois difficile à gérer lorsque la clé ne contient pas assez d'informations pour indiquer clairement à quelle année le mois appartient. Dans ce cas, cependant, la clé Month contient la clé Year qui à elle seule permet d'identifier les mois en tant que sous-groupe des années.

**Remarque** : Vous pouvez créer un déterminant qui regroupe les mois sans le contexte des années, mais ce choix est peu courant car le rapport correspondant contiendrait toutes les données du mois de février de toutes les années regroupées, au lieu des seules données de février 2006.

## Utilisation de déterminants avec des clés multiparties

Dans l'exemple de la dimension Time vu plus haut, une clé suffisait à identifier chaque ensemble de données pour un déterminant, mais ce n'est pas toujours le cas.

Par exemple, la dimension géographique ci-dessous utilise des définitions de clés multiparties pour tous les déterminants sauf un.

Région	Clé Region	Clé State/Province	Clé City
Amérique du Nord	Etats-Unis	Illinois	Springfield
Amérique du Nord	Etats-Unis	Missouri	Springfield
Amérique du Nord	Etats-Unis	Californie	Dublin
Europe	Irlande	n.d.	Dublin

A l'instar de l'exemple de la dimension Time, vous pouvez définir trois déterminants pour cet ensemble de données de la façon suivante : deux déterminants **Regroupement** (Region et State/Province) et un déterminant unique (City).

Nom du déterminant	Clé	Attributs	Identification de façon unique	Regroupement
Région	Clé Region	Aucun	Non	Oui
State/Province	Clé State/Province	Aucun	Non	Oui
Ville	Clé Region Clé State/Province Clé City	Aucun	Oui	Non

Dans ce cas, nous avons utilisé la clé Region, la clé State/Province et la clé City pour garantir l'unicité de City. Nous avons procédé ainsi parce que, dans les données fournies, certains noms de villes étaient communs à plusieurs états ou provinces, portant eux-mêmes des noms communs à plusieurs régions.

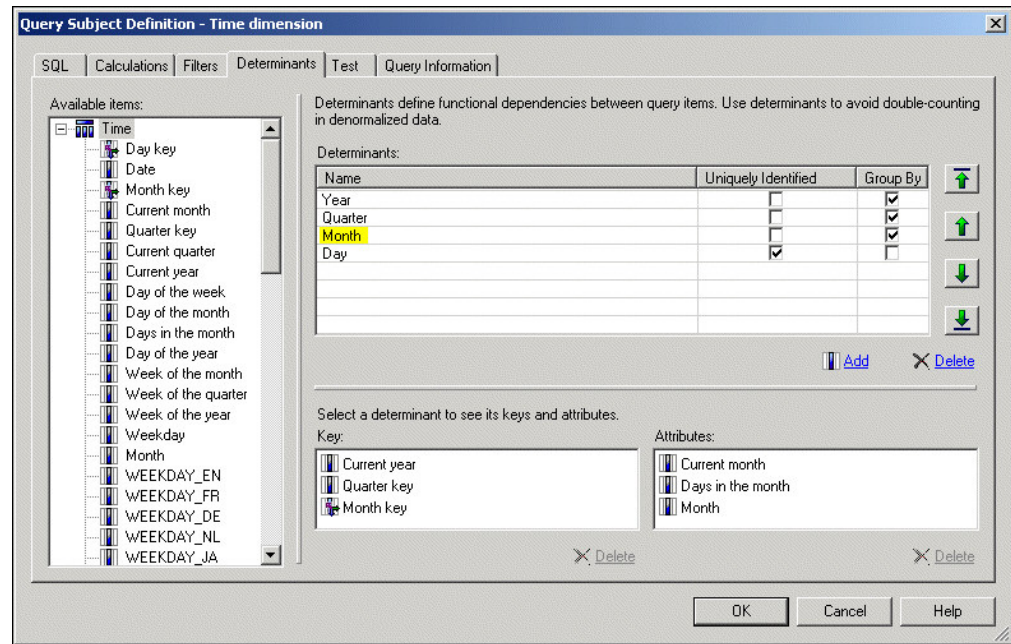
## Evaluation des déterminants dans leur ordre de spécification

Il n'existe pas de hiérarchie au sein des déterminants, mais un ordre d'évaluation s'applique. Lorsque le logiciel IBM Cognos examine une sélection d'éléments à partir d'un sujet de requête, il les compare à chaque déterminant (clés et attributs) un par un, selon l'ordre défini dans l'onglet **Déterminants**. De cette manière, le logiciel IBM Cognos sélectionne le déterminant qui correspond le mieux.

Dans l'exemple suivant, les attributs pour le mois en cours, les jours dans le mois et les noms de mois localisés sont associés à la clé Month. Lorsque la requête soumise fait référence à l'un ou l'autre de ces attributs, le déterminant Month est le premier déterminant sur lequel le critère de correspondance est satisfait. Si aucun autre attribut n'est requis, l'évaluation des déterminants s'arrête à Month et ce déterminant est utilisé pour les clauses group et for dans le code SQL.



Lorsque d'autres attributs de la dimension sont également inclus, mais sans avoir été mis en correspondance avec un déterminant précédent, le logiciel IBM Cognos continue son évaluation jusqu'à ce qu'une correspondance soit trouvée ou que le dernier déterminant soit atteint. C'est pour cette raison que tous les éléments de requête sont associés à un déterminant unique. Si aucune autre correspondance n'est trouvée, la clé unique de l'ensemble de données complet sert à déterminer la façon dont les données sont groupées.



## Situations dans lesquelles utiliser des déterminants

Les déterminants peuvent servir à résoudre différents problèmes liés à la granularité des données, mais vous devez les utiliser systématiquement dans les principaux cas suivants :

- Un sujet de requête qui se comporte comme une dimension à plusieurs niveaux de granularité et sera joint à des données de faits sur différents ensembles de clés.

Par exemple, la dimension Time a plusieurs niveaux et est jointe à la dimension Inventory sur la clé Month et à la dimension Sales sur la clé Day. Pour en savoir davantage, reportez-vous à la section «Requêtes à faits multiples, requêtes à granularité multiple», à la page 8.

- Il est nécessaire d'effectuer un comptage ou autres fonctions d'agrégation sur une clé ou un attribut qui sont répétés.

Par exemple, la dimension Time a une clé Month et un attribut Days in the month qui sont répétés pour chaque jour. Si vous voulez utiliser l'attribut Days in the month dans un rapport, vous ne souhaitez pas obtenir la somme des jours du mois pour chaque jour du mois. Au lieu de cela, vous voulez obtenir la valeur unique de Days in the month pour la clé Month choisie. Dans le code SQL, c'est-à-dire XMIN(Days in the month for Month\_Key). Il existe également une clause Group by dans le code Cognos SQL.

Il existe quelques cas moins fréquents dans lesquels il convient d'utiliser des déterminants :

- Vous souhaitez identifier de façon unique la ligne de données lors de l'extraction de données texte BLOB de la source de données.

L'utilisation de requêtes sur des données BLOB requiert des informations de clé ou d'index supplémentaires. Si ces informations ne sont pas disponibles dans la source de données, vous pouvez les ajouter à l'aide de déterminants. Remplacez les déterminants importés depuis la source de données qui sont en conflit avec des relations établies pour la génération de rapports.

Il n'est pas possible d'utiliser des clés à segments multiples lorsque le sujet de requête accède aux données BLOB. Avec des requêtes récapitulatives, les données BLOB doivent être extraites séparément de la partie récapitulative de la requête. Pour cela, vous avez besoin d'une clé identifiant de façon unique la ligne et n'étant pas associée à plusieurs segments.

- Il existe une jointure définie qui utilise moins de clés qu'un déterminant unique défini pour un sujet de requête.

Si votre jointure est basée sur un sous-ensemble des colonnes référencées par les clés d'un déterminant unique du côté 0..1 ou 1..1 des relations, un conflit se produit. Pour le résoudre, modifiez la relation pour être parfaitement conforme au déterminant ou modifiez le déterminant de façon à ce qu'il prenne en charge la relation.

- Vous souhaitez remplacer les déterminants importés depuis la source de données qui sont en conflit avec des relations établies pour la génération de rapports.

Par exemple, il existe des déterminants pour deux sujets de requête correspondant à plusieurs colonnes, mais la relation entre ces sujets de requête n'utilise qu'un sous-ensemble de ces colonnes. Modifiez les informations des déterminants du sujet de requête si elles ne sont pas adaptées à l'utilisation de colonnes supplémentaires dans la relation.

### **Requêtes à faits multiples, requêtes à granularité multiple**

Dans les sources de données relationnelles, des requêtes à granularité multiple et à faits multiples se produisent lorsqu'une table contenant des données dimensionnelles est jointe à plusieurs tables de faits sur différentes colonnes de clés.

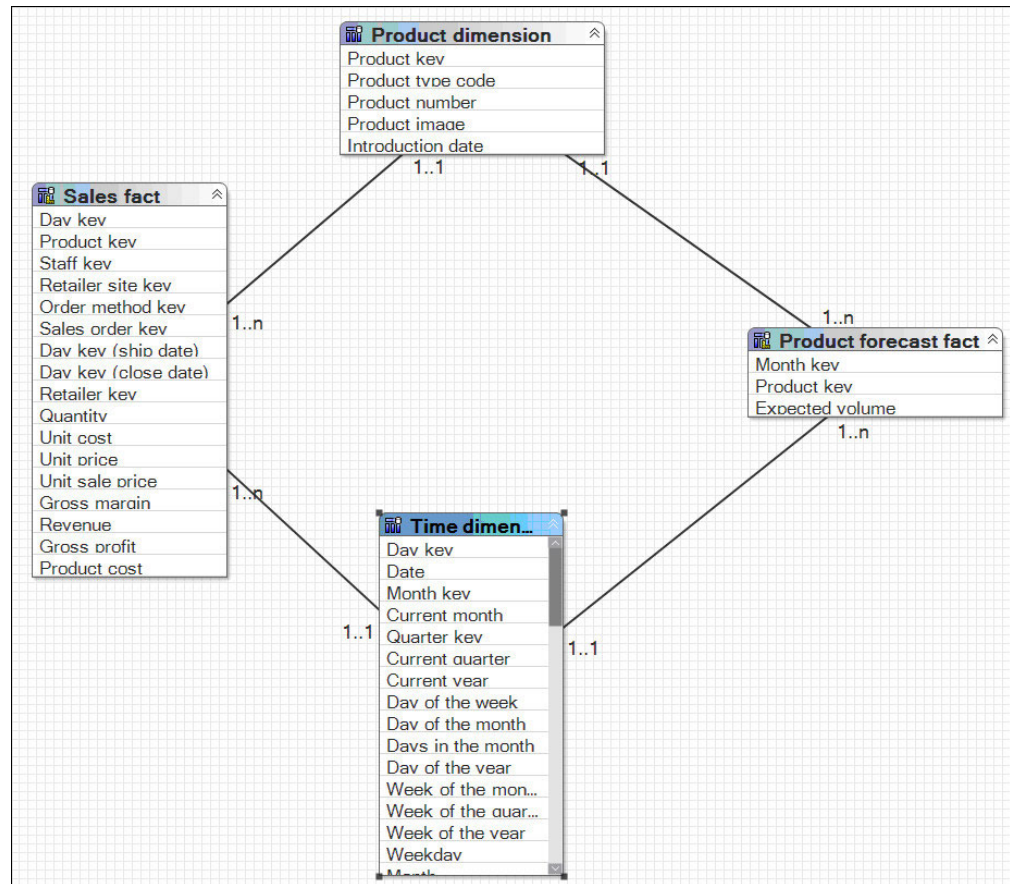
Notez que dans la présente section le terme dimension est utilisé au sens conceptuel. Un sujet de requête avec une cardinalité de 1:1 ou 0:1 se comporte comme une dimension. Pour en savoir davantage, reportez-vous à la section «Cardinalité», à la page 1.

Un sujet de requête a généralement des groupes ou niveaux distincts de données d'attributs avec des clés qui se répètent. Les studios d'IBM Cognos procèdent automatiquement à une agrégation au plus bas niveau commun de granularité dans le rapport. Le risque de double comptage apparaît lors de la création de totaux dans les colonnes contenant des données qui se répètent. Si le niveau de granularité des données est correctement modélisé, le double comptage peut être évité.

**Remarque :** Il est possible de consigner des données à un niveau de granularité inférieur au niveau commun le plus bas. Dans ce cas, les données d'un niveau de granularité plus élevé se répètent, mais cela n'a pas d'incidence sur les totaux si les déterminants sont correctement appliqués.

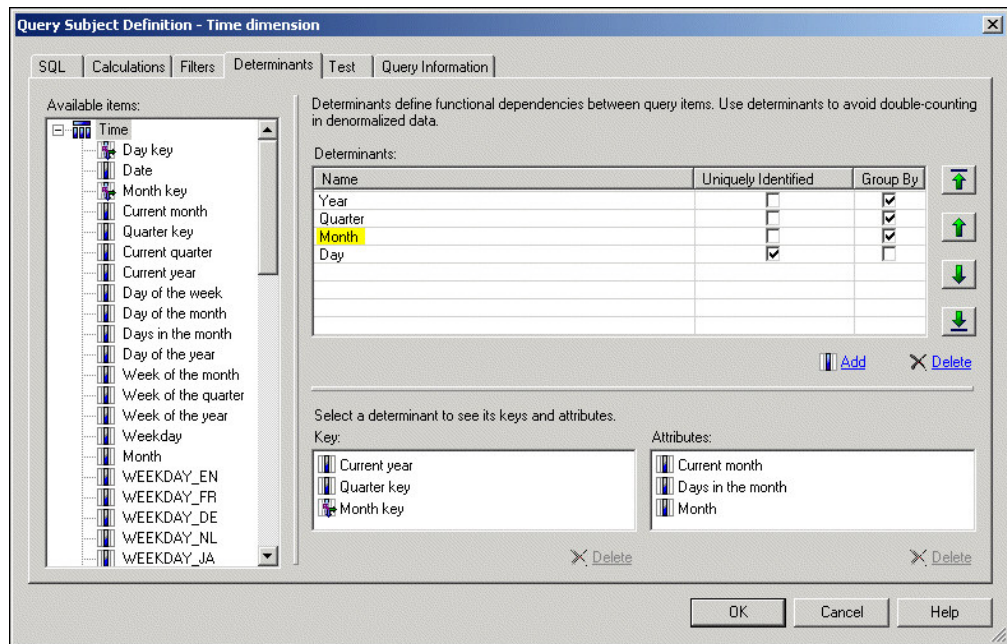


L'exemple ci-dessous montre deux sujets de requête de faits, Sales et Product forecast, qui partagent deux requêtes dimensionnelles ordinaires, Time et Product.



Dans cet exemple, la dimension Time représente le point central du problème de granularité. La dimension Sales est jointe à la dimension Time sur la clé Day et la dimension Product forecast est jointe à la dimension Time sur la clé Month. En raison des différentes clés de jointure, au moins deux déterminants doivent être clairement identifiés sur la dimension Time. Par exemple, les déterminants des clés de Month et Day sont identifiés. Day est la clé unique de la dimension Time, les clés Month sont répétées pour chaque jour du mois.

Par exemple, les déterminants de Month se présentent comme ceci :



Le sujet de requête Product pourrait avoir au moins trois déterminants : Product line, Product type et Product. Il dispose de relations avec les deux tables de faits sur la clé Product. Il n'existe aucun problème de granularité au niveau du sujet de requête Product.

Par défaut, un rapport est agrégé pour extraire des enregistrements de chaque table de faits au niveau de granularité commun le plus bas. Si vous créez un rapport utilisant Quantity dans Sales, Expected volume dans Product forecast, Month dans Time et Product name dans Product, le rapport extrait les enregistrements de chaque table de faits au niveau de granularité commun le plus bas. Dans l'exemple qui suit, il s'agit des niveaux Month et Product.

Pour éviter tout double comptage lorsqu'il existe des données à plusieurs niveaux de granularité, créez au moins deux déterminants pour le sujet de requête Time. Pour obtenir un exemple, reportez-vous à la section «Déterminants», à la page 4.

Mois	Nom de produit	Quantité	Volume prévu
Avril 2007	Calmo brûlures	1410	1690
Avril 2007	Parapluie Omnium	132	125
Février 2007	Calmo brûlures	270	245
Février 2007	Parapluie Omnium		1
Février 2006	Calmo brûlures	88	92

Si vous n'indiquez pas correctement les déterminants dans le sujet de requête Time, l'agrégation risque de ne pas être correcte. Par exemple, les valeurs Expected volume qui existent au niveau Month dans Product forecast sont répétées pour chaque jour dans le sujet de requête Time. Lorsque les déterminants ne sont pas définis correctement, les valeurs Expected volume sont multipliées par le nombre de jours dans le mois.

Mois	Nom de produit	Quantité	Volume prévu
Avril 2007	Calmo brûlures	1410	50 700
Avril 2007	Parapluie Omnium	132	3750
Février 2007	Calmo brûlures	270	7134
Février 2007	Parapluie Omnium		29
Février 2006	Calmo brûlures	88	2576

Vous remarquez que les chiffres de la colonne Expected volume sont différents.

## Considérations de conception de modèles

Lors de la construction d'un modèle, il est important de comprendre qu'il n'existe pas de flux de travaux unique pouvant fournir un modèle adapté à toutes les applications. Avant de commencer votre modèle, vous devez connaître les spécifications de l'application en matière de fonctionnalité, facilité d'utilisation et performances. La conception de la source de données et les spécifications de l'application vont déterminer les réponses à de nombreuses questions dans cette section.

### Où les relations et déterminants doivent-ils être créés ?

L'endroit où créer les relations fait partie des questions fréquemment posées. Les relations doivent-elles être créées entre les sujets de requête de source de données, entre les sujets de requête de modèle ou entre les deux ? La réponse peut varier car elle dépend de la complexité de la source de données que vous êtes en train de modéliser.

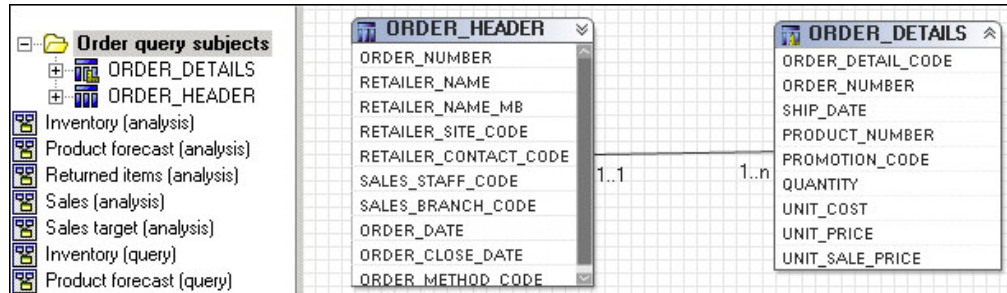
Lorsque vous utilisez des sujets de requête, les relations et déterminants s'appartiennent mutuellement.

Lorsque vous utilisez des sujets de requête de modèle, l'utilisation des relations et déterminants a certains effets secondaires.

- Le sujet de requête de modèle commence à fonctionner comme une vue, ce qui remplace le paramètre **As View** ou **Minimized** du type **SQL Generation** du sujet de requête.  
Cela signifie que le code SQL reste le même, quel que soit l'élément du sujet de requête auquel il est fait référence. Pour en savoir davantage, reportez-vous à la section «Définition du langage SQL réduit», à la page 12.
- Le sujet de requête de modèle devient un objet autonome.  
Cela signifie que les relations sous-jacentes ne sont plus appliquées, à l'exception de celles existant entre les objets référencés. Il faut parfois créer des relations supplémentaires pour palier celles qui étaient précédemment induites par les métadonnées des sujets de requête sous-jacents.
- Lorsqu'un déterminant est créé sur un sujet de requête de modèle, ce déterminant est ignoré tant qu'une relation n'est pas créée avec lui.

Voici un exemple de relation sur un sujet de requête de modèle qui remplace le paramètre **Minimized SQL** et simplifie le modèle. Dans cet exemple, Order Header et Order Details sont combinés de façon à se comporter comme un fait unique. Ils sont placés dans leur propre dossier et toutes leurs relations sont supprimées, à l'exception de la relation entre Order Header et Order Details.

C'est la seule relation qui sera effective une fois que le sujet de requête de modèle est créé et que des relations y sont rattachées.



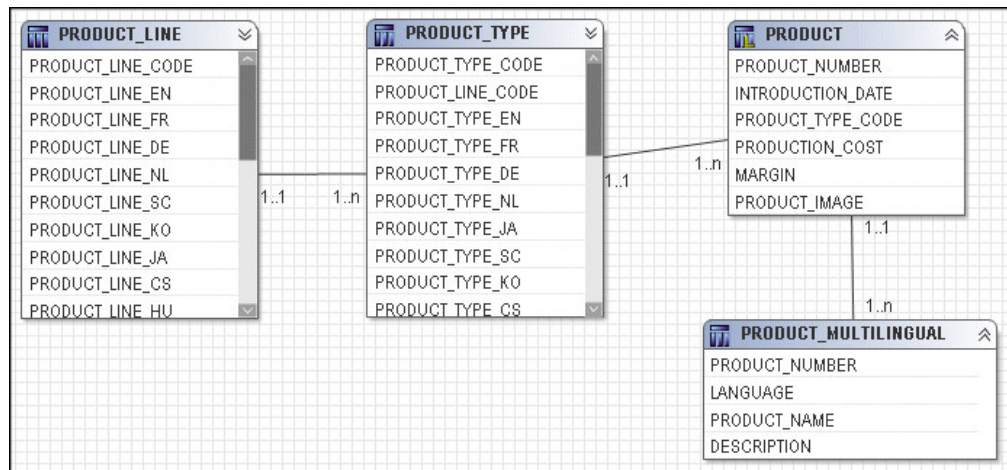
Pour choisir où définir les relations et déterminants dans le modèle, vous devez connaître l'impact du langage SQL réduit sur votre application.

Pour plus d'informations sur les relations, les déterminants et le code SQL réduit, voir les rubriques **Model Advisor** du *Guide d'utilisation d'IBM Cognos Framework Manager*.

### Définition du langage SQL réduit

Lorsque vous utilisez le langage SQL réduit, le code SQL généré contient uniquement l'ensemble de tables et de jointures minimum nécessaire à l'obtention des valeurs pour les éléments de requête sélectionnés.

Pour comprendre la signification du code SQL réduit, utilisez les tables Product suivantes. Quatre sujets de requête : Product Line, Product Type, Product et Product Multilingual sont tous joints les uns aux autres.



Ils peuvent être combinés dans un sujet de requête de modèle.

Products (model)
PRODUCT_LINE_CODE
PRODUCT_LINE
PRODUCT_TYPE_CODE
PRODUCT_TYPE
PRODUCT_NUMBER
PRODUCT_NAME
DESCRIPTION
PRODUCTION_COST
MARGIN
INTRODUCTION_DATE
PRODUCT_IMAGE

Si vous testez le sujet de requête de modèle Products dans son ensemble, vous pouvez voir que la clause from de la requête fait référence à quatre tables.

```
select
  PRODUCT_LINE.PRODUCT_LINE_CODE as Product_Line_Code,
  PRODUCT_LINE.PRODUCT_LINE_EN as Product_Line,
  PRODUCT_TYPE.PRODUCT_TYPE_CODE as Product_Type_Code,
  PRODUCT_TYPE.PRODUCT_TYPE_EN as Product_Type,
  PRODUCT.PRODUCT_NUMBER as Product_Number,
  PRODUCT_MULTILINGUAL.PRODUCT_NAME as Product_Name
  PRODUCT_MULTILINGUAL.DESCRPTION as Product_Description,
  PRODUCT.INTRODUCTION_DATE as Introduction_Date,
  PRODUCT.PRODUCT_IMAGE as Product_Image,
  PRODUCT.PRODUCTION_COST as Production_Cost,
  PRODUCT.MARGIN as Margin
from
  gos1_82..gos1.PRODUCT_LINE PRODUCT_LINE,
  gos1_82..gos1.PRODUCT_TYPE PRODUCT_TYPE,
  gos1_82..gos1.PRODUCT PRODUCT,
  gos1_82..gos1.PRODUCT_MULTILINGUAL PRODUCT_MULTILINGUAL
where
  (PRODUCT_MULTILINGUAL."LANGUAGE" - N'EN')
  and
  (PRODUCT_LINE.PRODUCT_LINE_CODE = PRODUCT_TYPE.PRODUCT_LINE_CODE)
  and
  (PRODUCT_TYPE.PRODUCT_TYPE_CODE = PRODUCT.PRODUCT_TYPE_CODE)
  and
  (PRODUCT.PRODUCT_NUMBER = PRODUCT_MULTILINGUAL.PRODUCT_NUMBER
```

Si vous testez uniquement Product name, vous pouvez voir que la requête qui en résulte utilise seulement Product Multilingual, qui est la table requise. Il s'agit des effets du SQL réduit.

```
select
  PRODUCT_MULTILINGUAL.PRODUCT_NAME as Product_Name
from
  gos1_82..gos1.PRODUCT_MULTILINGUAL PRODUCT_MULTILINGUAL
where
  (PRODUCT_MULTILINGUAL."LANGUAGE" - N'EN')
```

### Exemple : Lorsque le SQL réduit est nécessaire

Si vous modélisez une source de données normalisée, pensez au SQL réduit car il diminuera le nombre de tables utilisées dans certaines demandes et les performances s'en trouveront améliorées. Dans ce cas, il est préférable de créer des



relations et des déterminants entre les sujets de requête de la source de données, puis de créer des sujets de requête de modèle qui n'ont pas de relation.

Certains croient à tort qu'en l'absence de relation entre les objets, il est impossible de créer des groupes sous forme de schémas en étoile. Ce n'est pas le cas. Sélectionnez les sujets de requête de modèle à inclure dans le groupe et utilisez l'assistant **Star Schema Grouping**. Vous pouvez également créer des raccourcis et les déplacer dans un nouvel espace-noms. Les raccourcis vers les relations ne sont pas nécessaires ; cette fonctionnalité est purement visuelle dans le diagramme. L'effet est le même sur la génération des requêtes et la présentation dans les studios.

### **Exemple : Lorsque le SQL réduit n'est pas aussi important que les requêtes prévisibles**

Certains éléments d'une source de données doivent être encapsulés pour garantir qu'ils se comportent comme un objet de données unique. Il peut s'agir, par exemple, d'une table de sécurité qui doit être systématiquement jointe à un fait. Dans le modèle Ventes Vacances et Aventures, Order Header et Order Details constituent un jeu de tables représentant un fait et doivent toujours faire l'objet d'une requête conjointe. Pour obtenir un exemple, reportez-vous à la section «Où les relations et déterminants doivent-ils être créés ?», à la page 11.

### **Présentation de la mise en cache des métadonnées**

IBM Cognos Framework Manager stocke les métadonnées importées de la source de données. Cependant, selon les paramètres de restriction et certaines actions que vous effectuez dans le modèle, ces métadonnées peuvent ne pas être utilisées lors de la préparation d'une requête.

Si vous activez la restriction **Allow enhanced model portability at run time**, Framework Manager interroge systématiquement la source de données pour obtenir des informations sur les métadonnées avant de préparer une requête. Si vous n'avez pas activé cette restriction, Framework Manager va, dans la plupart des cas, accéder aux métadonnées qui ont été stockées dans le modèle au lieu d'interroger la source de données. Les principales exceptions sont les suivantes :

- Le code SQL d'un sujet de requête de source de données a été modifié. Ceci inclut l'utilisation de macros.
- Un calcul ou filtre a été ajouté à un sujet de requête de source de données.

**Remarque :** Les requêtes de métadonnées générées sont bien prises en charge par la plupart des fournisseurs de systèmes de gestion de bases de données relationnelles et ne devraient donc pas avoir une incidence notable sur la majeure partie des applications de génération de rapports.

### **Sujets de requête et dimensions**

Les sujets de requête et dimensions n'ont pas les mêmes objectifs. Le sujet de requête sert à générer des requêtes relationnelles et peut être créé à l'aide des règles du schéma en étoile, tandis que la dimension sert à la modélisation dimensionnelle des sources relationnelles, ce qui implique un comportement OLAP. Puisque les sujets de requête représentent la base des dimensions, la clé de la réussite pour tout modèle dimensionnel est un modèle relationnel sain.

Un modèle dimensionnel est nécessaire uniquement si vous voulez utiliser IBM Cognos Analysis Studio, afin d'activer les passages aux niveaux supérieurs et inférieurs dans les rapports, ou pour accéder aux fonctions membres dans les studios. Pour bon nombre d'applications, la fonctionnalité OLAP n'est pas

nécessaire. Par exemple, votre application est principalement destinée à la génération de rapports ou de requêtes ad hoc, sans qu'il soit nécessaire de passer au niveau supérieur ou inférieur, ou vous effectuez la maintenance d'un modèle n IBM Cognos ReportNet. Dans les deux cas, vous pouvez publier des packs basés uniquement sur des sujets de requête.

Les déterminants des sujets de requête sont à distinguer des niveaux et des hiérarchies des dimensions ordinaires, mais ils peuvent être étroitement liés à une hiérarchie unique. Si vous envisagez d'utiliser vos sujets de requête en tant que base de vos dimensions, vous devez réfléchir à la structure des hiérarchies que vous souhaitez créer et vous assurer que vous avez créé des déterminants qui vont produire des résultats corrects lors de l'agrégation. Assurez-vous de disposer des éléments suivants :

- Le sujet de requête doit avoir un déterminant défini pour chaque niveau de la hiérarchie dans la dimension ordinaire.
- Les déterminants doivent être définis dans le même ordre que les niveaux dans la dimension ordinaire.
- Si vous souhaitez plusieurs hiérarchies qui s'agrègent différemment, vous pouvez créer un sujet de requête supplémentaire avec d'autres déterminants en tant que source de l'autre hiérarchie.

La création d'un modèle relationnel complet qui fournit des résultats corrects et de bonnes performances vous garantit une bonne base pour développer un modèle dimensionnel. En outre, en vous assurant qu'une couche d'objets de modèle (des sujets de requête ou des dimensions) existe entre la source de données et les objets visibles dans les studios, il vous est plus facile de protéger vos utilisateurs des modifications.

### **Objets de modèle et raccourcis**

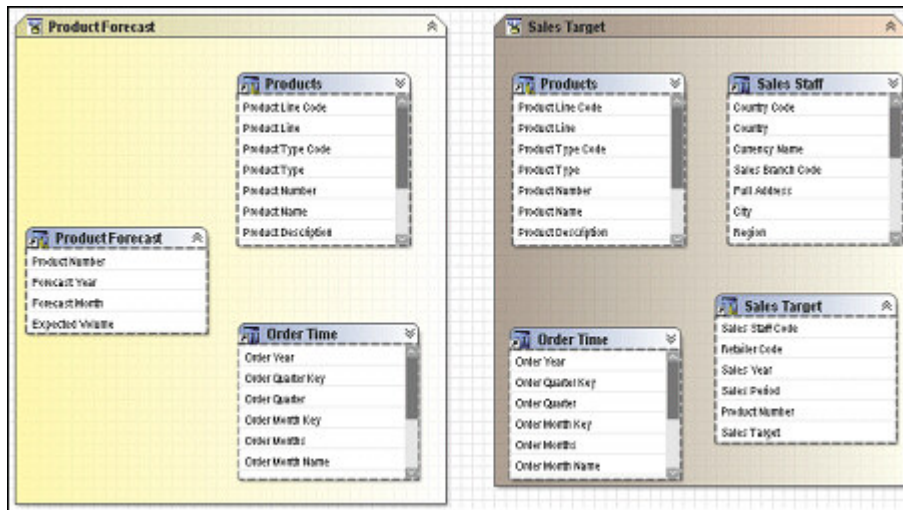
La principale différence entre les objets de modèle et les raccourcis est que les objets de modèle vous laissent la liberté d'inclure ou d'exclure des éléments et de les renommer. Vous pouvez utiliser des objets de modèle au lieu de raccourcis si vous devez limiter les éléments de requête à inclure ou renommer les éléments.

Les raccourcis sont moins souples que les objets de modèle du point de vue de la présentation, mais leur maintenance est beaucoup plus légère car ils sont automatiquement mis à jour dès que l'objet cible est mis à jour. Si la maintenance est l'une de vos principales préoccupations et que vous n'avez pas besoin de personnaliser l'apparence du sujet de requête, utilisez des raccourcis.

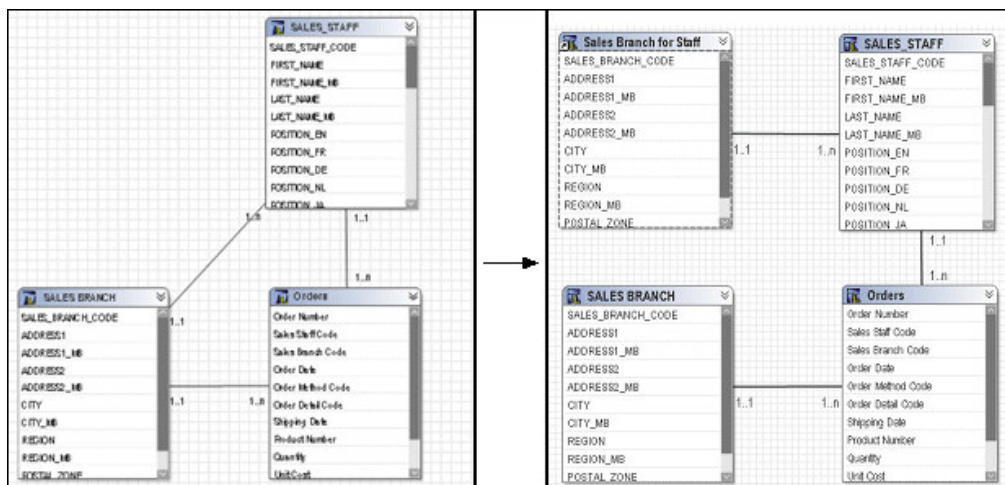
IBM Cognos Framework Manager comporte deux types de raccourcis :

- les raccourcis ordinaires, qui sont une simple référence à l'objet cible.
- les raccourcis d'alias, qui se comportent comme s'ils étaient une copie de l'objet d'origine avec un comportement complètement indépendant. Les raccourcis d'alias ne sont disponibles que pour les sujets de requête et les dimensions.

Les raccourcis ordinaires sont généralement utilisés en tant que dimensions conformes avec groupes sous forme de schémas en étoile, en créant plusieurs références avec exactement les mêmes nom et apparence à plusieurs endroits. Dans l'exemple ci-dessous, les raccourcis créés pour Products et Order Time se comportent comme des références. Si une requête est écrite pour obtenir Products à partir de Product Forecast et Sales Target, la requête utilise la définition de Products basée sur le raccourci d'origine et cette définition figure une seule fois dans la requête.



Les raccourcis d'alias sont généralement utilisés dans les dimensions de rôle ou les tables partagées. Il y a déjà un exemple relatif aux dimensions de rôle dans le présent document, nous allons donc étudier le cas des tables partagées. Dans cet exemple, Sales Staff et Sales Branch peuvent être considérés comme des hiérarchies distinctes. Notre connaissance des données nous permet de savoir que les membres du personnel peuvent changer de succursale, il faut donc que nous puissions produire des rapports sur Sales Branch et Sales Staff à la fois indépendamment et de façon groupée. Pour ce faire, nous devons créer un alias de Sales Branch pouvant être utilisé en tant que niveau dans la hiérarchie Sales Staff.



Une fois le nouveau raccourci d'alias en place, il est possible de créer des requêtes qui demandent les commandes par succursale et les commandes par équipe commerciale, avec en même temps les informations relatives à leur succursale en cours.

Lorsque vous ouvrez un modèle d'une édition précédente, la restriction **Shortcut Processing** est définie sur **Automatic**. Lorsque l'option **Automatic** est utilisée, les raccourcis fonctionnent de la même façon que dans les éditions précédentes, c'est-à-dire qu'un raccourci qui se trouve dans le même répertoire que sa cible se comporte comme un alias, ou une instance indépendante, alors qu'un raccourci qui se trouve ailleurs dans le modèle se comporte comme une référence au raccourci d'origine. Pour tirer parti de la propriété **Treat As**, il est conseillé de vérifier le



modèle et de définir la restriction à **Explicit** lors de la réparation. La réparation définit tous les raccourcis à la valeur correcte à partir de la propriété **Treat As** selon les règles suivies par le paramètre **Automatic**. Cela signifie que le comportement de votre modèle ne changera pas tant que vous ne modifierez pas la propriété **Treat As** de vos raccourcis.

Lorsque vous créez un modèle, la restriction **Shortcut Processing** est toujours définie à **Explicit**.

Lorsque la restriction est définie à **Explicit**, le comportement du raccourci est dicté par la propriété **Treat As** et vous avez un contrôle total sur la façon dont les raccourcis se comportent sans avoir à connaître leur emplacement dans le modèle.

## Dossiers et espaces-noms

Il est important de savoir à propos des espaces noms, qu'une fois la création du rapport commencée, tout changement de nom d'un espace noms publié aura une incidence sur votre contenu IBM Cognos. En effet, le changement du nom de l'espace-noms modifie les identificateurs des objets publiés dans les métadonnées. L'espace noms fait partie de l'identificateur de l'objet dans IBM Cognos Framework Manager, chaque espace noms doit donc avoir un nom unique dans le modèle. Chaque objet de l'espace-noms doit également avoir un nom unique. Une partie de la stratégie d'élaboration des groupes sous forme de schémas en étoile est de placer les raccourcis dans un espace-noms distinct, ce qui crée automatiquement un identificateur unique pour chaque objet dans l'espace-noms. Pour les bases de données relationnelles, ceci vous permet d'utiliser le même nom pour les raccourcis vers des dimensions conformes dans différents groupes sous forme de schémas en étoile.

Lors de la prochaine tentative d'exécuter une requête, un rapport ou une analyse sur le modèle mis à jour, vous obtenez une erreur. Si vous devez renommer l'espace-noms que vous avez publié, utilisez la commande **Analyze Publish Impact** pour déterminer les rapports sur lesquels cela aura une incidence.

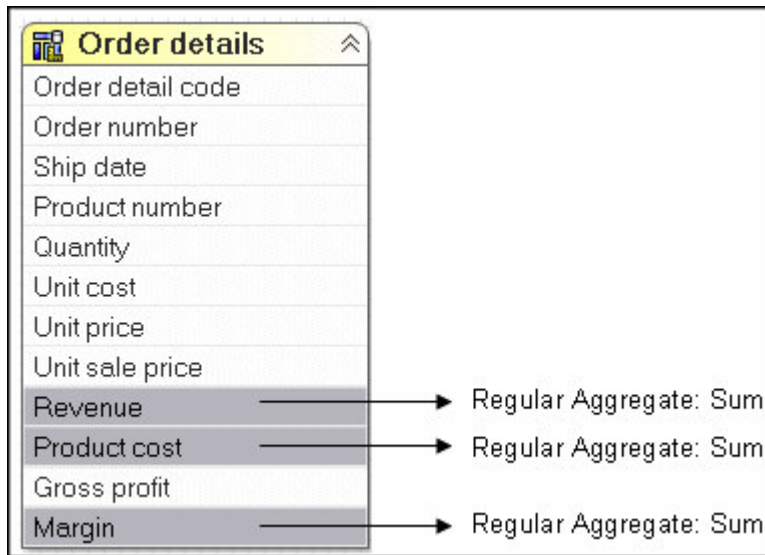
Les dossiers sont beaucoup moins compliqués que les espaces-noms. Ils n'ont qu'une fonction purement organisationnelle et n'ont aucune incidence sur l'identificateur des objets ou sur votre contenu. Vous pouvez créer des dossiers pour organiser les objets par sujet ou domaine fonctionnel. Cela facilite l'accès aux métadonnées, en particulier dans les projets volumineux.

Le principal inconvénient des dossiers est qu'ils nécessitent des noms uniques pour tous les sujets de requête, dimensions et raccourcis. Ils ne représentent donc pas la solution idéale pour contenir des objets partagés.

## Définition de l'ordre des opérations pour les calculs de modèles

Dans certains cas (notamment pour les calculs liés à un ratio), il est utile de procéder à l'agrégation sur les termes du calcul avant d'effectuer l'opération mathématique.

Par exemple, le fait Order details suivant contient des informations sur chaque commande :



L'élément Margin calcule la proportion de bénéfices :

$$\text{Margin} = (\text{Revenue} - \text{Product cost}) / \text{Revenue}$$

Si vous exécutez une requête pour afficher les éléments Revenue, Product cost et Margin de chaque produit en utilisant le fait Order details, vous obtenez les résultats suivants :

Numéro de produit	Revenu	Coût de produit	Marge
1	23 057 141 \$	11 292 005 \$	61038 %
2	11 333 518 \$	6 607 904 \$	49606 %

Vous pouvez constater que la valeur Margin paraît fausse. Cela est dû à l'ordre des opérations utilisées pour calculer cet élément. L'élément Margin est calculé de la façon suivante :

$$\text{Margin} = \text{sum}(\text{Revenue} - \text{Product cost}) / \text{Revenue}$$

L'agrégation a eu lieu après l'opération mathématique et, dans ce cas, les résultats obtenus sont erronés.

Pour obtenir les résultats souhaités pour Margin, vous devez effectuer l'agrégation avant l'opération mathématique :

$$\text{Margin} = (\text{sum}(\text{Revenue}) - \text{sum}(\text{Product cost})) / \text{sum}(\text{Revenue})$$

Vous obtenez alors les résultats suivants :

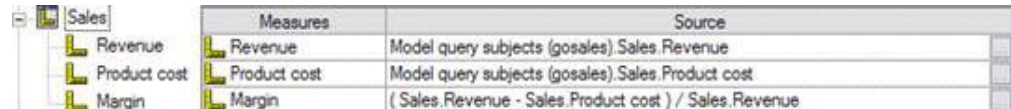
Numéro de produit	Revenu	Coût de produit	Marge
1	23 057,141 \$	11 292 005 \$	51,03 %
2	11 333 518 \$	6 607 904 \$	41,70 %

Cette opération peut être réalisée dans IBM Cognos Framework Manager en créant un calcul autonome pour Margin et en définissant sa propriété **Regular Aggregate** sur **Calculated**. Chaque élément de requête figurant dans l'expression du calcul est agrégé conformément à la propriété **Regular Aggregate**. Les propriétés **Regular**

**Aggregate** des éléments Revenu et Product cost sont définies sur **Sum**, c'est pourquoi, lors du calcul, la somme est utilisée pour agréger ces termes.

**Remarque :** Le type d'agrégation calculée n'est pas pris en charge par les calculs imbriqués dans des sujets de requête. Il est uniquement pris en charge par les calculs autonomes ou par ceux imbriqués dans des dimensions de mesure et basés sur des mesures issues de la même dimension.

Examinez par exemple le calcul Margin imbriqué dans la dimension de mesure Sales :



	Measures	Source
Sales	Revenue	Model query subjects (gosales).Sales.Revenue
	Product cost	Model query subjects (gosales).Sales.Product cost
	Margin	( Sales.Revenue - Sales.Product cost ) / Sales.Revenue

Dans cet exemple, l'élément Margin est basé sur les mesures Product cost et Revenu, qui font partie de la même dimension de mesure, à savoir Sales. Si la propriété **Regular Aggregate** de l'élément Margin est définie sur **Calculated**, celui-ci est cumulé de la façon suivante :

$$\text{Margin} = \text{sum}(\text{Revenue} - \text{Product cost}) / \text{sum}(\text{Revenue})$$

Si Margin repose sur les éléments de requête source des mesures Product cost et Revenu (Sales (model).Product cost, Sales (model).Revenue), l'agrégation calculée n'est pas prise en charge et le comportement d'agrégation automatique est utilisé. Dans ce cas, l'élément Margin est cumulé de la façon suivante :

$$\text{Margin} = \text{sum}(\text{Revenue} - \text{Product cost}) / \text{Revenue}$$

Pour plus d'informations sur la modification du mode d'agrégation des éléments de requête, voir le *Guide d'utilisation d'IBM Cognos Framework Manager*.

## Incidence de la taille du modèle

La taille du modèle peut avoir une incidence sur l'efficacité de l'application Framework Manager.

Les modèles particulièrement volumineux entraînent des temps de traitement plus longs et, dans des cas extrêmes, il peut même arriver que la mémoire soit alors insuffisante. Les actions telles que Analyze Publish Impact, Find Report Dependencies, Publish Packages et Run Model Advisor s'exécutent de manière optimale sur des modèles ne dépassant pas 50 mégaoctets.

## Concepts de modélisation dimensionnelle

Les dimensions ordinaires et de mesure permettent l'activation d'une présentation OLAP des métadonnées, les passages au niveau supérieur et inférieur ainsi qu'une variété de fonctions OLAP. Vous devez utiliser des groupes sous forme de schémas en étoile (un fait avec plusieurs dimensions) pour utiliser Analysis Studio d'IBM Cognos Analysis Studio avec une source de données relationnelle.

Lors de la construction du modèle, il est conseillé de créer les dimensions ordinaires de modèle et les dimensions de mesure de modèle sur la base d'un modèle relationnel dans lequel des concepts de schéma en étoile ont été appliqués.

Vous pouvez convertir des sujets de requête de source de données en dimensions de source de données, mais ces dernières ont des fonctionnalités limitées par

rapport aux sujets de requête ou dimensions de modèle et ne sont pas recommandées pour une utilisation générale.

## Dimensions ordinaires

Les dimensions ordinaires représentent des données descriptives qui fournissent du contexte aux données modélisées dans des dimensions de mesure. La dimension ordinaire est divisée en groupes d'informations appelés niveaux. A leur tour, les différents niveaux peuvent être organisés en hiérarchies. Par exemple, une dimension de produit peut contenir les niveaux Product Line, Product Type et Product dans une hiérarchie unique appelée Product. Autre exemple, une dimension de temps comprenant les niveaux Year, Quarter, Month, Week et Day organisés en deux hiérarchies. La première hiérarchie, YQMD, contient les niveaux Year, Quarter, Month et Day, tandis que la seconde hiérarchie YWD contient les niveaux Year, Week et Day.

La définition la plus simple d'un niveau est une clé métier et une légende, chacun d'eux faisant référence à un élément de requête. Une instance (ou ligne) d'un niveau est définie comme membre de ce niveau. Elle est identifiée par un nom unique de membre, lequel contient les valeurs des clés métier du niveau en cours et des niveaux supérieurs. Par exemple, [gosa]es].[Products].[ProductsOrg].[Product]->[All Products].[1].[1].[2] identifie un membre qui se trouve au quatrième niveau, Product, de la hiérarchie ProductsOrg de la dimension [Products] qui est dans l'espace-noms [gosa]es. La légende de ce produit est Gamelle ChefDeCamp, qui est le nom affiché dans l'arborescence de métadonnées et dans le rapport.

Le niveau peut être défini comme unique si la clé métier du niveau est suffisante pour identifier chaque ensemble de données pour un niveau. Dans le modèle Ventes Vacances et aventure, la définition de Product Type n'est pas nécessaire pour les membres du niveau Product car aucun numéro de produit n'est affecté à différents types de produit. Un niveau qui n'est pas défini comme unique est similaire à un déterminant qui utilise des clés multiparties, car des clés de niveaux de granularité supérieures sont nécessaires. Voir «Utilisation de déterminants avec des clés multiparties», à la page 6. Si des membres ne sont pas uniques au sein de membres ancêtres mais que le niveau est défini comme unique, les données des membres non uniques font l'objet d'un rapport sous la forme d'un seul membre. Par exemple, si le membre City est défini comme unique et identifié par un nom, les données relatives à Londres en Angleterre et à Londres au Canada seront combinées.

## Dimensions de mesure

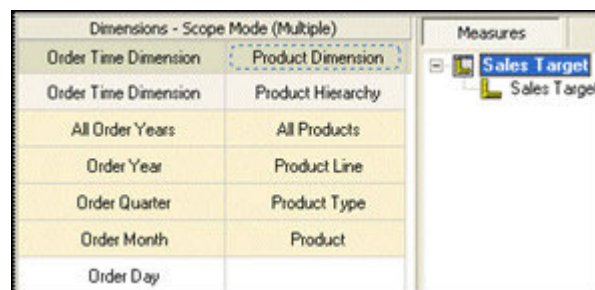
Les dimensions de mesure représentent les données quantitatives décrites par des dimensions ordinaires. Connue sous plusieurs noms dans différents produits OLAP, la dimension de mesure est tout simplement l'objet contenant les données de fait. Les dimensions de mesure sont différentes des sujets de requête de fait car elles n'incluent pas les clés étrangères utilisées pour joindre un sujet de requête de fait à un sujet de requête dimensionnel. En effet, la dimension de mesure n'est pas destinée à être jointe comme s'il s'agissait d'un objet de données relationnel. Pour la génération des requêtes, la dimension de mesure obtient sa relation à partir d'une dimension ordinaire par le biais des sujets de requête sous-jacents. De même, la relation vis-à-vis des autres dimensions de mesure s'effectue pour des dimensions ordinaires basées sur des sujets de requête construits de façon à se comporter comme des dimensions conformes. Pour activer les requêtes à granularité multiple et à faits multiples, vous devez disposer de sujets de requête et de déterminants créés de façon appropriée avant de construire des dimensions ordinaires et des dimensions de mesure.

## Relations de portée

Les relations de portée sont établies uniquement entre des dimensions de mesure et des dimensions ordinaires pour définir le niveau auquel les mesures sont disponibles à des fins de génération de rapports. Elles sont différentes des jointures et n'ont aucune incidence sur la clause Where. Dans une relation de portée, aucune condition ni critère ne sont définis sur la façon de former une requête, il y est uniquement indiqué si une dimension peut faire l'objet de requêtes avec un fait spécifique. L'absence de relation de portée peut entraîner une erreur lors de l'exécution ou provoquer un cumul des données de fait à un niveau supérieur à celui prévu, en raison des autres éléments du rapport.

Si vous définissez la relation de portée pour la dimension de mesure, les mêmes paramètres s'appliquent à toutes les mesures de la dimension de mesure. Si les données font l'objet de rapports à un niveau différent des mesures dans la dimension de mesure, vous pouvez définir la portée d'une mesure. Vous pouvez définir le niveau le plus bas à partir duquel les données peuvent faire l'objet d'un rapport.

Dans cet exemple, la dimension de mesure Sales Target a une seule mesure comprenant le niveau Order Month dans la dimension Order Time et le niveau Product de la dimension Product. Cela signifie que, lorsque vos utilisateurs tenteront d'accéder au détail au-delà du niveau des mois, ils verront des données répétées.



The screenshot shows the 'Dimensions - Scope Mode (Multiple)' window in IBM Cognos Framework Manager. It displays two dimensions: 'Order Time Dimension' and 'Product Dimension'. The 'Order Time Dimension' is expanded to show its hierarchy: 'All Order Years', 'Order Year', 'Order Quarter', 'Order Month', and 'Order Day'. The 'Product Dimension' is also expanded to show its hierarchy: 'All Products', 'Product Line', 'Product Type', and 'Product'. To the right, the 'Measures' window shows a single measure named 'Sales Target'.

Dimensions - Scope Mode (Multiple)		Measures
Order Time Dimension	Product Dimension	Sales Target
Order Time Dimension	Product Hierarchy	Sales Target
All Order Years	All Products	
Order Year	Product Line	
Order Quarter	Product Type	
Order Month	Product	
Order Day		

---

## Construction du modèle relationnel

La modélisation dimensionnelle de sources de données relationnelles est possible dans IBM Cognos Framework Manager, mais elle requiert un modèle relationnel bien construit.

IBM Cognos ReportNet offrait des fonctionnalités dimensionnelles permettant la création de requêtes à faits multiples et empêchant le double comptage. Succédant à IBM Cognos ReportNet, le logiciel IBM Cognos offre des fonctionnalités conçues explicitement pour la représentation dimensionnelle des métadonnées et des capacités OLAP avec les sources de données relationnelles. Les concepts appliqués à la modélisation relationnelle dans IBM Cognos ReportNet sont maintenus même s'ils ont subi quelques modifications, documentées dans le manuel *Framework Manager User Guide*.

Lorsque vous créez un nouveau modèle Framework Manager, vous devez suivre une procédure commune pour définir la génération de requêtes, même si vous n'avez pas l'intention d'utiliser les fonctionnalités de modélisation dimensionnelle. Vous devez modéliser de façon dimensionnelle une source de données relationnelle pour pouvoir l'utiliser dans IBM Cognos Analysis Studio, afin d'activer les passages aux niveaux supérieurs et inférieurs dans les rapports, ou pour accéder aux fonctions membres dans les studios.

## Définition de la base de la modélisation relationnelle

Un modèle est l'ensemble des objets connexes requis par une ou plusieurs applications de génération de rapports associées. Un modèle relationnel sain est la base de tout modèle dimensionnel.

Lorsque vous définissez la base du modèle relationnel, vous devez :

- Importation des métadonnées. Pour plus d'informations sur l'importation, voir le *Guide d'utilisation d'IBM Cognos Framework Manager*.
- «Vérification des métadonnées importées».
- «Résolution des relations ambiguës», à la page 23.
- Simplification du modèle relationnel à l'aide de concepts de schémas en étoile par l'analyse de la cardinalité des faits et des dimensions et le choix de l'emplacement des relations et des déterminants «Considérations de conception de modèles», à la page 11.
- Ajoutez des données de sécurité, si nécessaire. Pour en savoir davantage sur les données de sécurité, reportez-vous au manuel Framework Manager *User Guide*.

Vous pouvez alors définir la représentation dimensionnelle du modèle , si nécessaire, et organiser le modèle pour la présentation.

### Vérification des métadonnées importées

Après avoir importé des métadonnées vous devez les vérifier.

Vérifiez les zones suivantes :

- relations et cardinalité,
- déterminants,
- propriété **Usage** des éléments de requêtes,
- propriété **Regular Aggregate** des éléments de requêtes.

Les notions de relation et de cardinalité sont traitées ci-après. Pour en savoir davantage sur les propriétés **Usage** et **Regular Aggregate**, reportez-vous au manuel Framework Manager *User Guide*.

### Analyse de la cardinalité des faits et dimensions :

La cardinalité d'une relation définit le nombre de lignes d'une table liées aux lignes d'une autre table sur la base d'un ensemble (ou jointure) spécifique de clés. Dans IBM Cognos la cardinalité sert à déduire quels sont les sujets de requête qui se comportent comme des faits ou des dimensions. Il en résulte qu'IBM Cognos peut automatiquement résoudre une forme commune de jointure de boucle causée par des données de schéma en étoile, lorsque plusieurs tables de faits sont jointes à un ensemble commun de tables de dimensions.

Pour garantir une génération prévisible de requêtes, vous devez comprendre la façon dont la cardinalité est utilisée afin de l'appliquer correctement dans votre modèle. Il est conseillé d'observer le schéma de la source de données sous-jacente et de modifier les zones où la cardinalité identifie de façon incorrecte les faits ou dimensions, pouvant ainsi provoquer une génération imprévisible des requêtes. Vous pouvez utiliser la fonction **Model Advisor** dans Framework Manager pour comprendre la manière dont la cardinalité est interprétée.

Pour en savoir davantage, reportez-vous à la section «Cardinalité», à la page 1.



## Résolution des relations ambiguës

Les relations ambiguës se produisent lorsque les données représentées par un sujet de requête ou une dimension sont visibles dans au moins deux contextes ou rôles ou qu'elles peuvent être jointes de plusieurs façons.

Les relations ambiguës les plus fréquentes sont :

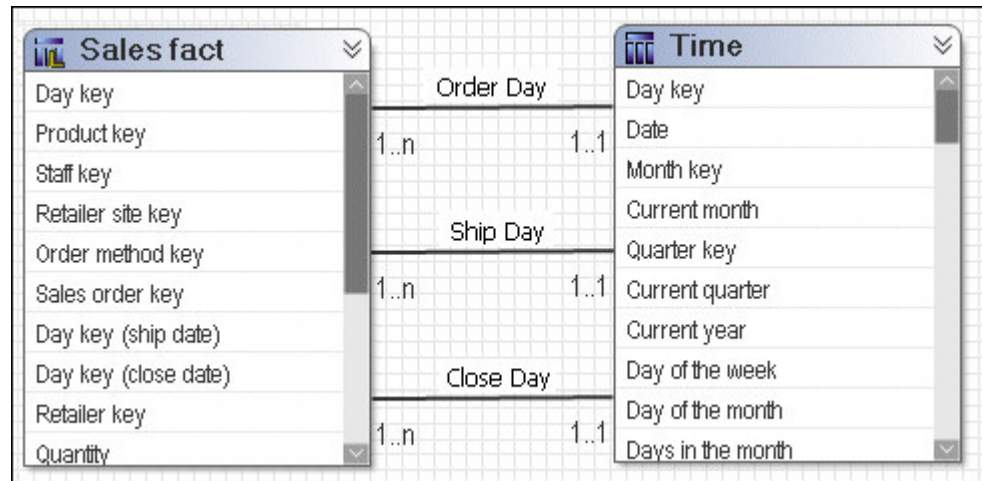
- «Dimensions de rôles»
- «Jointures de boucles», à la page 26
- «Relations réflexives et récursives», à la page 27

Vous pouvez utiliser **Model Advisor** pour détecter les relations qui pourraient poser des problèmes de génération de requêtes et les résoudre d'une des façons décrites ci-dessous. Notez qu'il existe d'autres façons de résoudre les problèmes abordés ici. L'objectif principal est de simplifier les chemins d'accès de requête.

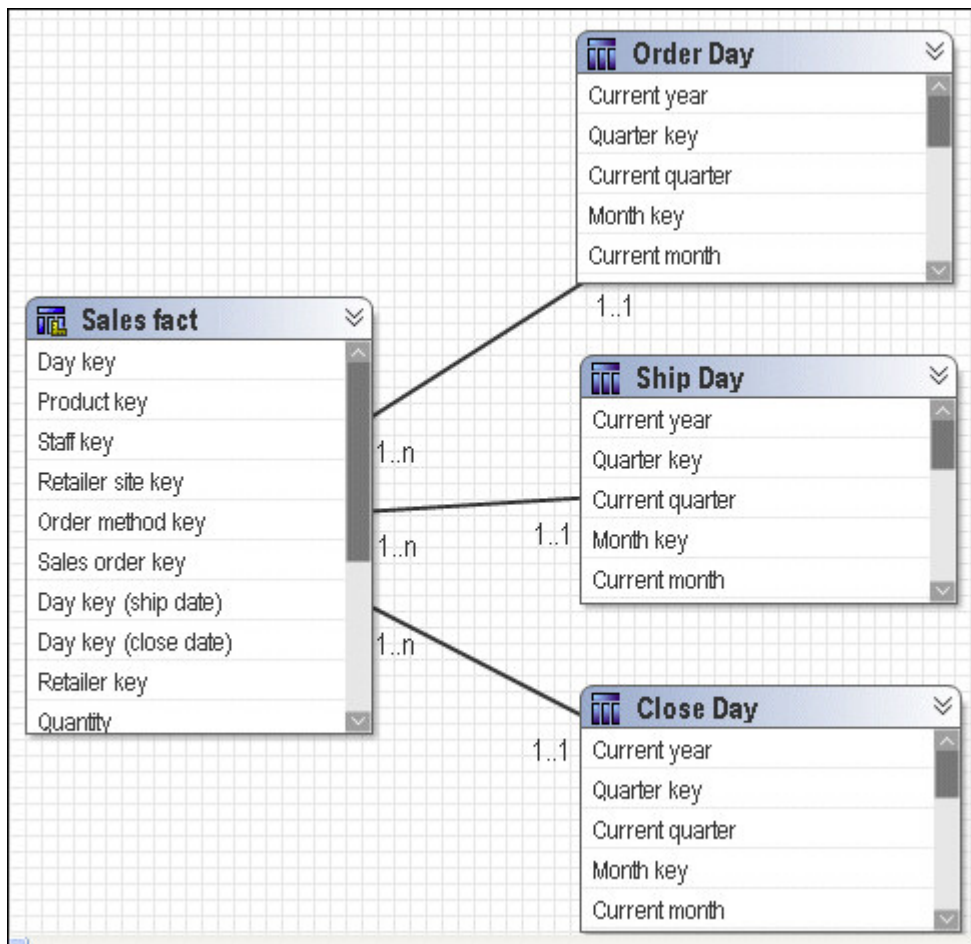
### Dimensions de rôles :

Une table présentant plusieurs relations valides entre elle et une autre table est appelée dimension de rôles. Cela se produit la plupart du temps avec des dimensions telles que Time et Customer.

Par exemple, il existe plusieurs relations entre le fait Sales et le sujet de requête Time par le biais des clés Order Day, Ship Day et Close Day.



Supprimez les relations des objets, sujets de requête de faits et sujets de requête de dimensions de rôle importés. Créez un sujet de requête de modèle pour chaque rôle. Eventuellement, pensez à exclure les éléments de requête superflus pour réduire la longueur de l'arborescence de métadonnées affichée pour les utilisateurs. Assurez-vous qu'il existe une relation appropriée unique entre chaque sujet de requête de modèle et le sujet de requête de fait. **Remarque** : Ceci va remplacer le paramètre **Minimized SQL** mais, compte tenu de la représentation à une seule table de la dimension Time, ce n'est pas un problème dans le cas présent.

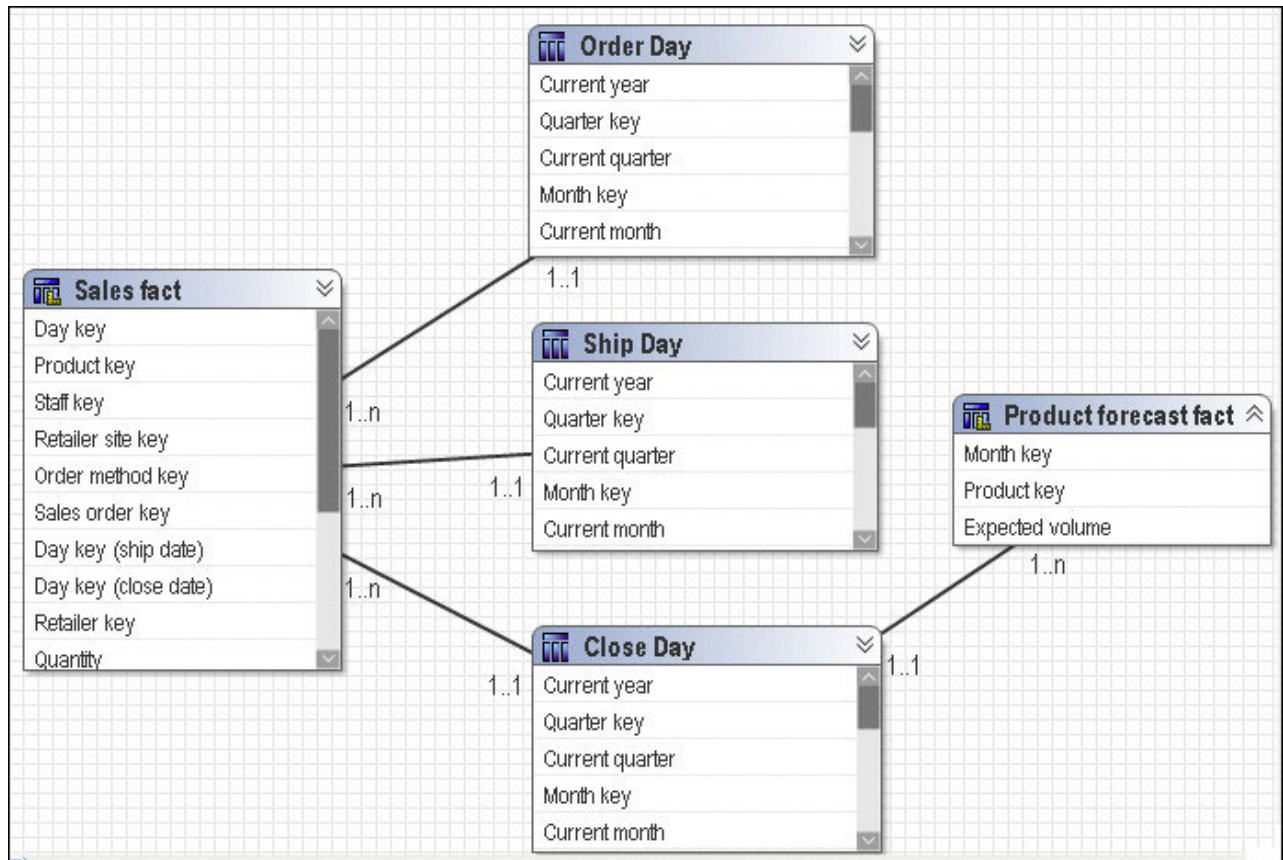


Indiquez de quelle façon utiliser ces rôles avec d'autres faits ne partageant pas les mêmes concepts. Par exemple, le fait Product forecast n'est associé qu'à une seule clé Time. Vous devez connaître vos données et votre activité pour déterminer si l'ensemble ou une partie des rôles créés pour la dimension Time sont applicables au fait Product forecast.

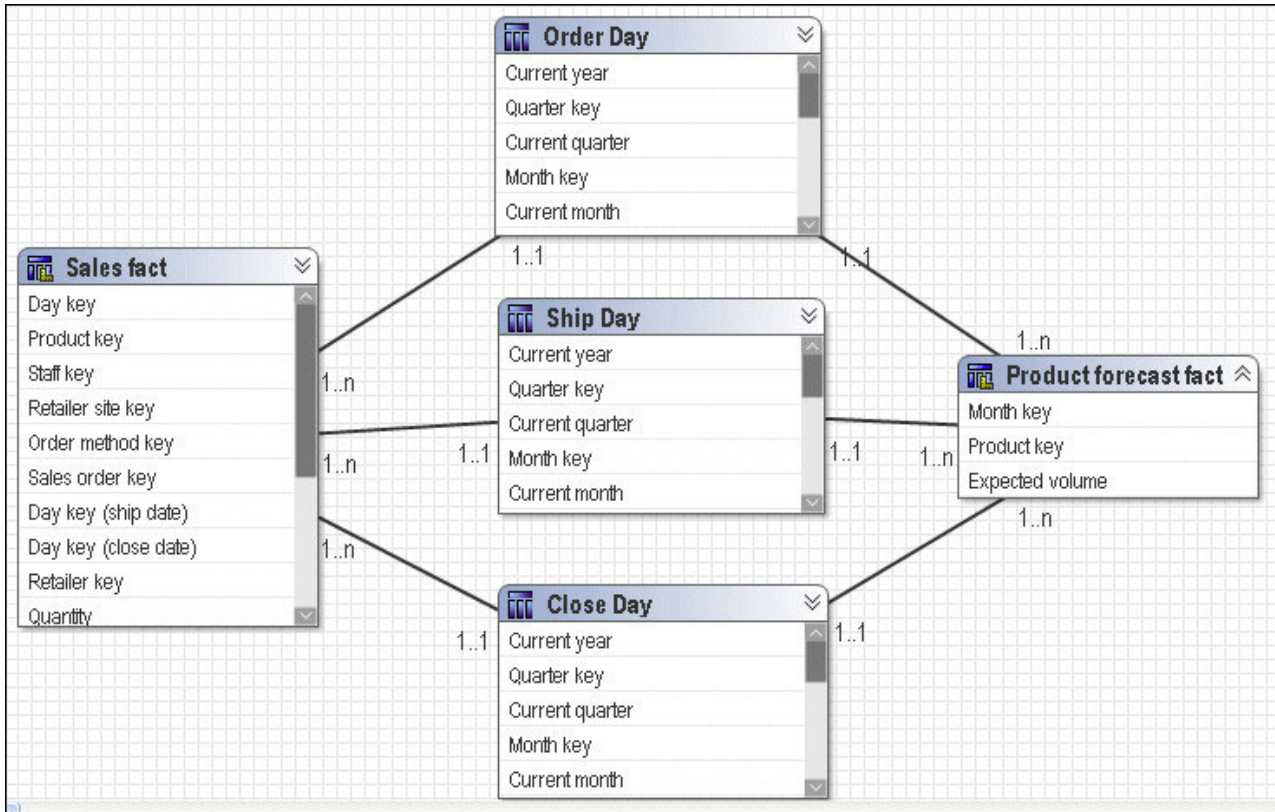
Dans cet exemple, vous pouvez effectuer l'une des opérations suivantes :

- Créer un sujet de requête supplémentaire en tant que dimension de temps conforme et le nommer de façon explicite en tant que dimension conforme. Choisir le rôle que vous allez utiliser le plus. Vous pouvez ensuite vérifier qu'une jointure est établie entre cette version et tous les faits la requérant. Dans cet exemple, c'est Close Day qui a été choisi.





- Considérer les éléments Ship Day, Order Day et Close Day comme des sujets de requête de date interchangeables avec le fait Product forecast. Dans ce cas, vous devez créer des jointures entre chaque dimension de rôles et le fait Product forecast. Lorsque vous émettez une requête concernant le fait Product forecast, vous ne pouvez utiliser qu'une seule dimension de temps à la fois, sinon votre rapport risque de ne contenir aucune donnée. Par exemple, Month\_key=Ship Month Key (200401) et Month key=Close Month Key (200312).



Si vous procédez à une modélisation dimensionnelle, utilisez chaque sujet de requête de modèle en tant que source de dimension ordinaire, puis nommez la dimension et les hiérarchies de façon appropriée. Vérifiez qu'il existe une relation de portée correspondante spécifique à chaque rôle.

#### Jointures de boucles :

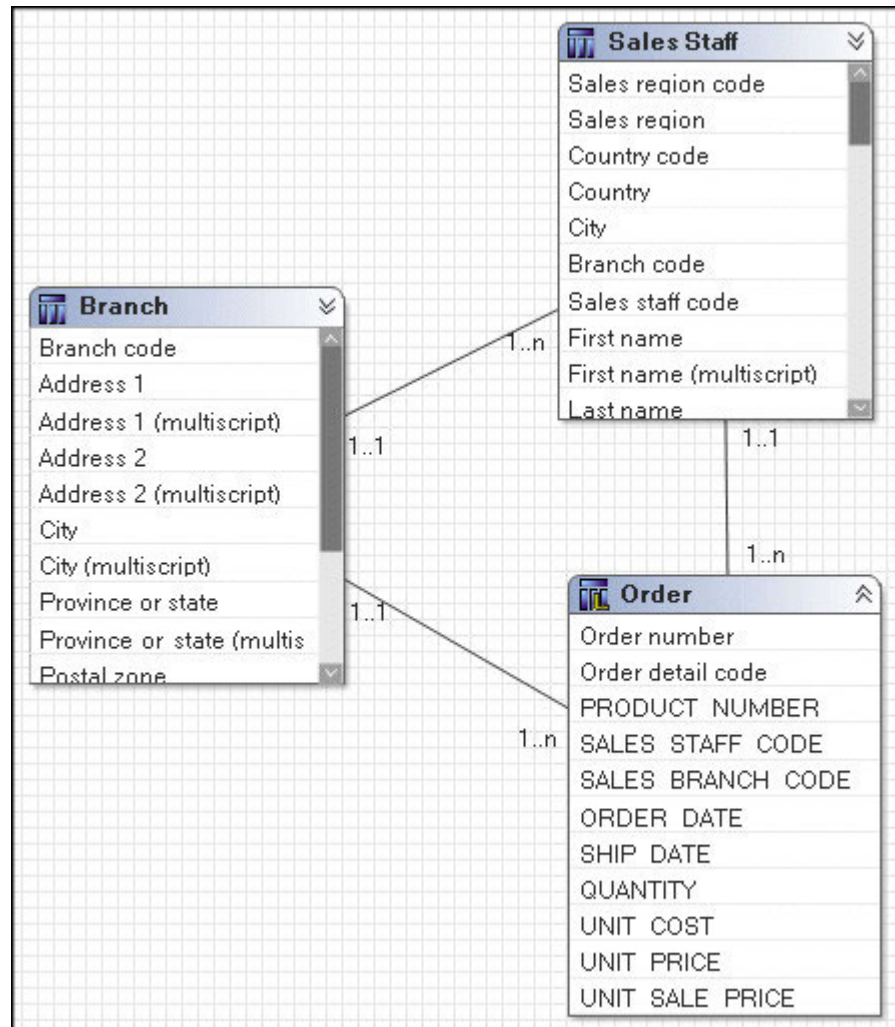
Les jointures de boucles du modèle sont généralement source de comportement imprévisible. Ce n'est pas le cas des jointures de boucles de schémas en étoile.

**Remarque :** Si la cardinalité identifie clairement les faits et dimensions, le logiciel IBM Cognos peut automatiquement résoudre les jointures de boucles causées par des données de schémas en étoile lorsque plusieurs tables de faits sont jointes à un ensemble commun de tables de dimensions.

En cas de jointures de boucles, les premières manifestations du problème sont des sujets de requête définis de façon ambiguë. Lorsque les sujets de requête sont définis de façon ambiguë et font partie d'une jointure de boucle, les jointures utilisées dans une requête donnée sont choisies selon différents facteurs, tels que l'emplacement des relations, le nombre de segments dans les chemins de jointure et, si aucune différence n'est trouvée dans ces éléments, le premier chemin de jointure par ordre alphabétique. Ceci est source de confusion chez vos utilisateurs et nous vous conseillons d'identifier clairement les chemins de jointure dans votre modèle.

Sales Staff et Branch offrent un bon exemple de jointure de boucle avec sujets de requête définis de façon ambiguë.

Dans cet exemple, il est possible de joindre Branch soit directement à Order soit par l'intermédiaire de Sales Staff. Le principal problème est que le résultat obtenu n'est pas le même entre le chemin de jointure Branch / Order et le chemin de jointure Branch / Sales Staff / Order. Effectivement, les employés peuvent changer de succursale pendant l'année. Ceux qui l'ont fait font l'objet d'un cumul dans leur nouvelle succursale, même si une grande partie de leurs ventes avaient été effectuée dans leur succursale précédente. Compte tenu de la conception du modèle, il n'y a aucun moyen de savoir lequel des deux chemins de jointure sera utilisé, cela peut varier en fonction des éléments choisis dans la requête.



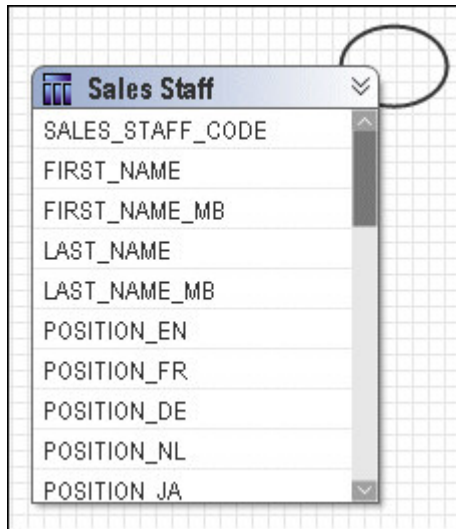
### Relations réflexives et récursives :

Les relations réflexives et récursives impliquent un minimum de deux niveaux de granularité. IBM Cognos Framework Manager importe les relations réflexives mais ne les utilise pas lors de l'exécution de requêtes. Les relations réflexives, qui correspondent à des jointures automatiques, sont affichées dans le modèle à titre représentatif uniquement.

Pour créer une relation réflexive opérationnelle, vous pouvez créer un raccourci d'alias, une copie du sujet de requête de la source de données ou un sujet de requête de modèle. Vous devez ensuite créer une relation entre le sujet de requête d'origine et le nouveau. L'utilisation d'un sujet de requête de modèle s'avère plus adaptée en matière de souplesse, puisque vous pouvez préciser quels éléments de

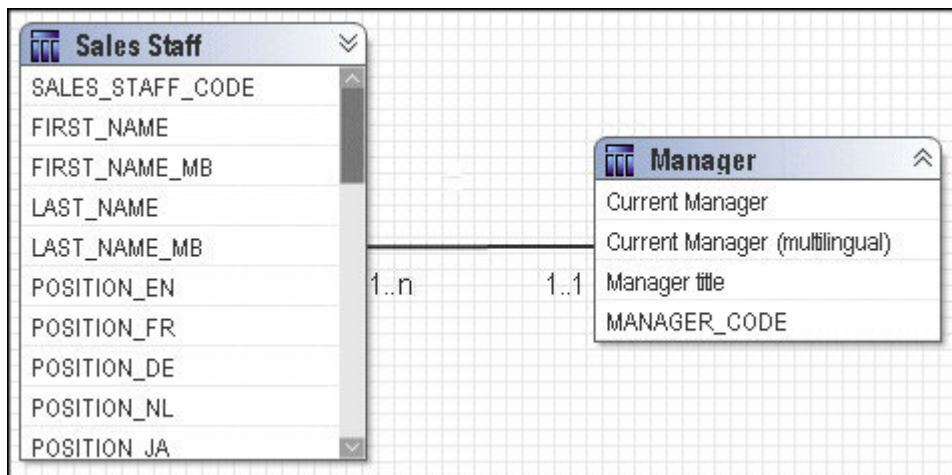
requêtes doivent être inclus dans le sujet de requête. Les raccourcis sont une meilleure solution du point de vue de la maintenance. Pour en savoir davantage, reportez-vous à la section «Objets de modèle et raccourcis», à la page 15.

Par exemple, le sujet de requête Sales Staff présente une relation récursive entre les éléments Sales\_Staff\_Code et Manager\_Code.



Créez un sujet de requête de modèle pour représenter la catégorie Manager. Créez une relation de type 1..1 à 1..n entre les catégories Manager et Sales Staff. Puis fusionnez-les dans un nouveau sujet de requête de modèle.

Pour une structure simple à deux niveaux utilisant un sujet de requête de modèle Manager basé sur la catégorie Sales Staff, le modèle se présente comme suit :



Pour une hiérarchie récursive et équilibrée, répétez cette opération pour chaque niveau supplémentaire de la hiérarchie.

Dans le cas d'une hiérarchie non équilibrée ou récursive possédant un grand nombre de niveaux, il est conseillé d'aplanir la hiérarchie dans la source de données et de modéliser la hiérarchie ainsi aplanie en une seule dimension ordinaire.

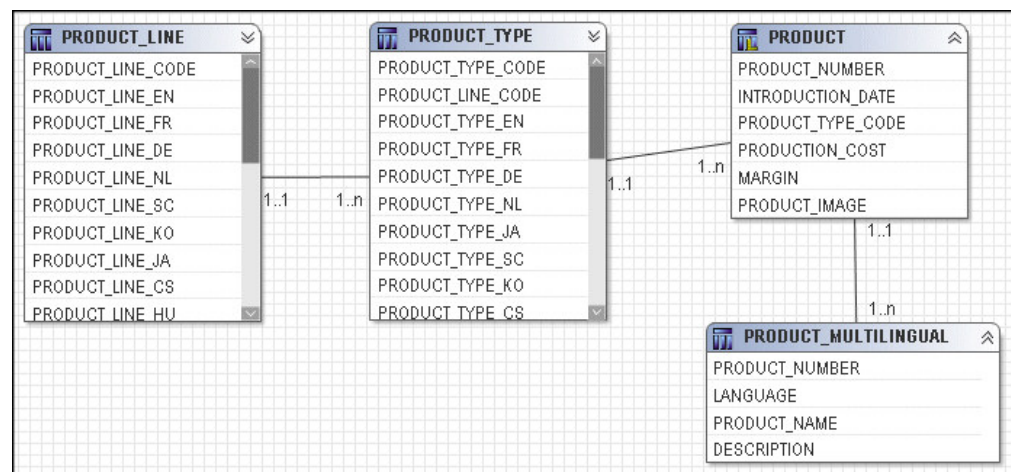
## Simplification du modèle relationnel

Vous pouvez simplifier le modèle en appliquant des concepts de schéma en étoile aux données dimensionnelles et aux données de fait.

### Modélisation des sujets de requête représentant des données descriptives :

La modélisation dimensionnelle du logiciel IBM Cognos nécessite l'application des principes du schéma en étoile aux couches logiques du modèle.

Les sources de données normalisées ou de type Flocon de neige sont souvent dotées de plusieurs tables décrivant un concept professionnel unique. Par exemple, une représentation normalisée du concept Product peut inclure quatre tables liées par des relations de type 1..n. Chaque catégorie Product Line est constituée d'une ou de plusieurs catégories Product Types. Chaque catégorie Product Type contient une ou plusieurs catégories Products. Les produits sont associés à des noms et des descriptions dans plusieurs langues, ils existent donc dans la table de recherche Product Multilingual.



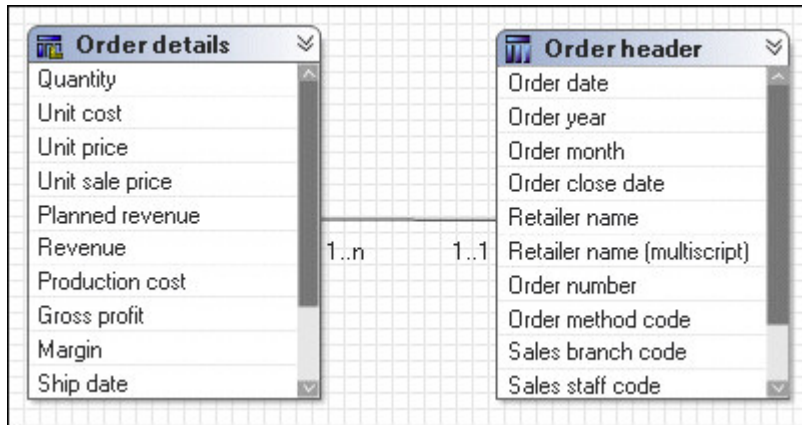
Une façon de simplifier le modèle consiste à créer un sujet de requête de modèle pour chaque concept métier descriptif. Il est possible que vos utilisateurs ne connaissent pas la relation entre les sujets de requête individuels, il est donc utile de les regrouper ; en outre, devoir développer chaque objet de modèle et sélectionner un élément de requête demande plus d'efforts.

L'étape suivante de l'analyse consiste à créer une dimension ordinaire avec un niveau pour chaque sujet de requête.

### Modélisation de données de fait :

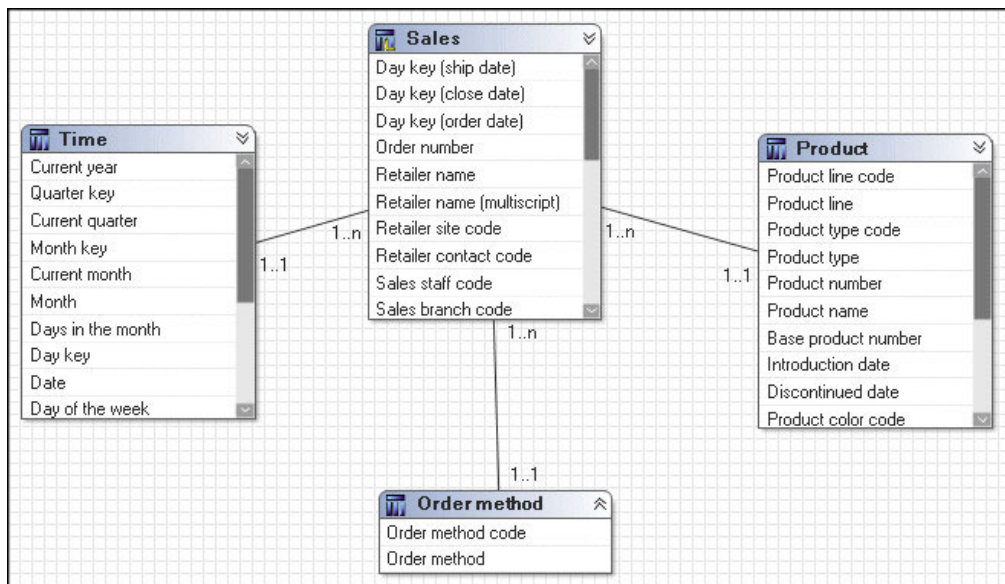
Les sources de données comportent souvent des tables de type Principale/Détails, contenant elles-mêmes des faits. Par exemple, lorsque les tables Order header et Order details sont utilisées pour insérer et mettre à jour des données, la structure Principale/Détails présente des avantages. Lorsque ces tables sont utilisées pour la génération de rapports et l'analyse de données, vous pouvez les regrouper en un seul concept métier logique de façon à simplifier le modèle. Vous pouvez également insérer une dimension entre elles, Returned Items, par exemple. Choisissez la solution qui correspond le mieux à vos besoins.





Pour simplifier le modèle de cet exemple, appliquez des concepts de schéma en étoile afin de créer un sujet de requête de modèle combinant les clés étrangères des tables Order header et Order details et incluant toutes les mesures au niveau Order details. Ce sujet de requête peut être joint aux mêmes sujets de requête que ceux auxquels Order header et Order details ont été joints. Vous pouvez supprimer les relations d'origine des deux sujets de requête de source de données, à l'exception de la relation qui définit la jointure entre eux. Pour connaître les avantages et les inconvénients de la création de relations vis-à-vis de sujets de requête, consultez les exemples fournis dans la section «Définition du langage SQL réduit», à la page 12.

Dans l'exemple ci-dessous, Order Header et Order Details ont été combinés dans un nouveau sujet de requête de modèle nommé Sales. Ce sujet de requête a été joint à Product, Time et Order method.



L'étape suivante de l'analyse consiste à créer une dimension de mesure basée sur le sujet de requête de modèle.

## Définition de la représentation dimensionnelle du modèle

La modélisation dimensionnelle de sources de données relationnelles est une des fonctions offertes par IBM Cognos Framework Manager. Vous pouvez modéliser des dimensions avec des hiérarchies et des niveaux et avoir des faits dotés de plusieurs mesures. Vous pouvez ensuite relier les dimensions aux mesures en définissant la portée du modèle.

Vous devez modéliser de façon dimensionnelle une source de données relationnelle pour pouvoir l'utiliser dans IBM Cognos Analysis Studio, afin d'activer les passages aux niveaux supérieurs et inférieurs dans les rapports, ou pour accéder aux fonctions membres dans les studios.

Vous pouvez utiliser le modèle relationnel comme couche de base puis définir la représentation dimensionnelle du modèle.

Vous pouvez ensuite organiser le modèle pour la présentation. Voir «Organisation du modèle», à la page 34.

### Création de dimensions ordinaires

Une dimension ordinaire contient des informations de clé d'affaires et des informations descriptives classées selon une hiérarchie, du niveau de granularité le plus élevé au plus faible. Elle dispose généralement de plusieurs niveaux qui doivent tous avoir une clé et une légende. Si vous n'avez pas de clé unique pour votre niveau, il est conseillé d'en créer une dans un calcul.

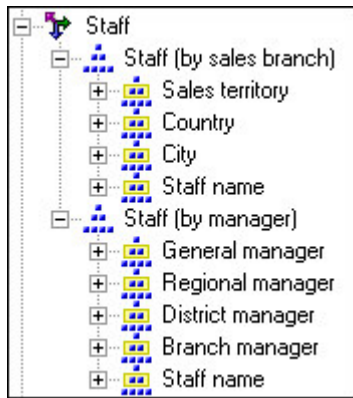
Les dimensions ordinaires de modèle sont basées sur des sources de données ou des sujets de requête de modèle déjà définis dans le modèle. Vous devez définir une clé métier et une légende de type de chaîne pour chaque niveau. Lorsque vous vérifiez le modèle, toute absence de clé métier et de légende est détectée. Au lieu de joindre des dimensions ordinaires de modèle aux dimensions de mesure, créez des jointures sur les sujets de requête sous-jacents et créez une relation de portée entre la dimension ordinaire et la dimension de mesure.

### Modélisation de dimensions avec plusieurs hiérarchies

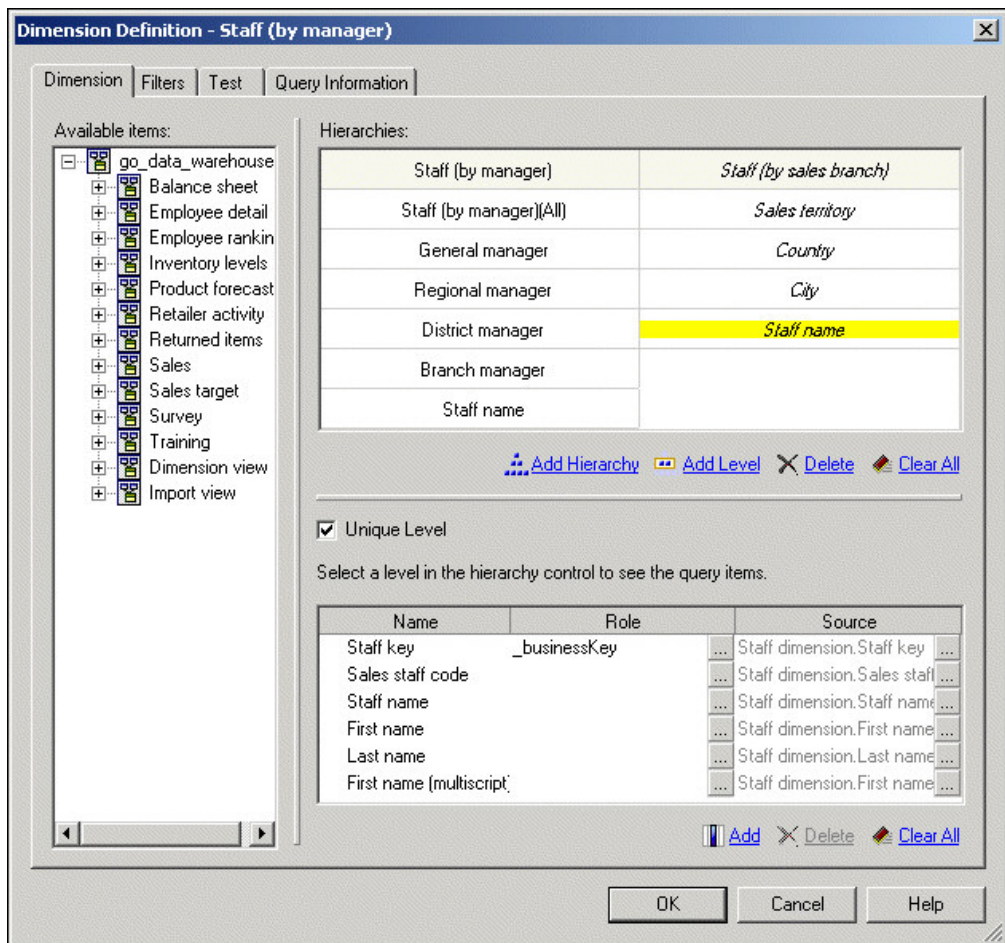
Des hiérarchies multiples sont créées lorsque différentes vues structurées peuvent s'appliquer aux mêmes données. Selon la nature des hiérarchies et des rapports requis, vous aurez peut-être besoin de vérifier la technique de modélisation appliquée, selon le cas.

Par exemple, l'équipe commerciale (Sales Staff) peut être visualisée par responsable ou par zone géographique. Dans les studios d'IBM Cognos ces hiérarchies sont des structures logiques séparées, mais interchangeable, liées à la même requête sous-jacente.

Voici une dimension unique d'équipe commerciale avec deux hiérarchies :



Les hiérarchies sont définies comme suit dans Framework Manager :

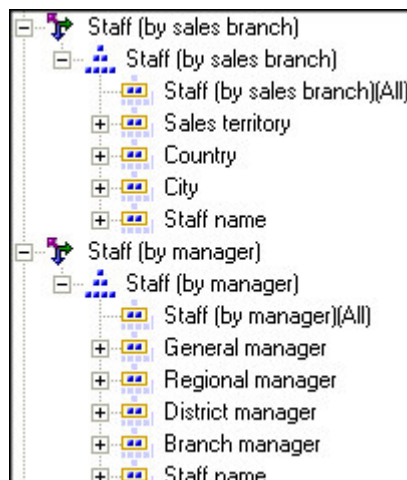


Dans Framework Manager, vous pouvez indiquer des hiérarchies multiples pour des dimensions ordinaires. Les hiérarchies multiples d'une dimension ordinaire se comportent comme des vues d'une même requête. Néanmoins, vous ne pouvez utiliser qu'une seule hiérarchie à la fois dans une requête. Par exemple, vous ne pouvez pas utiliser une hiérarchie sur les lignes d'un rapport de type tableau croisé et une autre hiérarchie de la même dimension dans les colonnes. Si vous avez besoin de ces deux hiérarchies dans le même rapport, vous devez créer deux dimensions, une pour chaque hiérarchie. Si vous disposez de plusieurs hiérarchies avec des niveaux ou agrégations sensiblement différents, vous pouvez effectuer



voire modélisation de façon à ce qu'un sujet de requête distinct, avec les déterminants appropriés, serve de base à cette hiérarchie. La seule obligation est de définir une jointure entre le sujet de requête utilisé en tant que base de la hiérarchie et le sujet de requête qui fournit les données de faits.

Voici des dimensions distinctes pour chaque hiérarchie :



Utilisez cette approche si des ensembles de colonnes sensiblement différents correspondent à chaque hiérarchie et s'il s'avère plus intuitif pour vos utilisateurs de modéliser les hiérarchies en tant que dimensions distinctes, avec des requêtes distinctes et plus simples.

## Création de dimensions de mesure

Une dimension de mesure est une série de faits. Vous pouvez créer une dimension de mesure pour un ou plusieurs sujets de requête entre lesquels il existe une relation valide.

Les dimensions de mesure de modèle doivent être exclusivement composées d'éléments quantitatifs. Par nature, il n'existe pas de clés sur lesquelles appliquer la jointure, il est donc impossible de créer des jointures vers des dimensions de mesure de modèle. Au lieu de joindre de dimensions de mesure de modèle aux dimensions ordinaires, créez des jointures au niveau des sujets de requête sous-jacents. Ensuite, créez manuellement une relation de portée entre eux ou bien détectez cette portée si les deux dimensions se trouvent dans le même espace-noms.

## Création de relations de portée

Les relations de portée sont établies uniquement entre des dimensions de mesure et des dimensions ordinaires pour définir le niveau auquel les mesures sont disponibles à des fins de génération de rapports. Elles sont différentes des jointures et n'ont aucune incidence sur la clause Where. Dans une relation de portée, aucune condition ni critère ne sont définis sur la façon de former une requête, il y est uniquement indiqué si une dimension peut faire l'objet de requêtes avec un fait spécifique. En l'absence de relation de portée, une erreur se produit lors de l'exécution.

Si vous définissez la relation de portée pour la dimension de mesure, les mêmes paramètres s'appliquent à toutes les mesures de la dimension de mesure. Si les

données font l'objet de rapports à un niveau différent des mesures dans la dimension de mesure, vous pouvez définir la portée d'une mesure. Vous pouvez définir le niveau le plus bas à partir duquel les données peuvent faire l'objet d'un rapport.

Lorsque vous créez une dimension de mesure, IBM Cognos Framework Manager crée une relation de portée entre la dimension de mesure et chaque dimension ordinaire existante. Framework Manager recherche les chemins de jointure entre la dimension de mesure et les dimensions ordinaires, en commençant par le niveau de détail le plus bas. Si de nombreux chemins de jointure sont disponibles, la relation de portée créée par Framework Manager peut ne pas être celle prévue. Dans ce cas, vous devez éditer la relation de portée.

## Organisation du modèle

Après avoir travaillé sur la base du modèle relationnel et créé une représentation dimensionnelle vous pouvez organiser le modèle.

- Conservez les métadonnées de la source de données dans un espace-noms ou dossier distinct.
- Créez au moins un espace-noms ou dossier facultatif pour résoudre les difficultés relatives à la requête à l'aide de sujets de requête.

Pour utiliser IBM Cognos Analysis Studio, il doit y avoir un espace noms ou dossier dans le modèle qui représente les métadonnées avec objets dimensionnels.

- Créez au moins un espace-noms ou dossier pour la vue métier globale des métadonnées contenant des raccourcis vers les dimensions ou sujets de requête. Utilisez des concepts commerciaux pour modéliser la vue métier. Un même modèle peut contenir de nombreuses vues métier, chacune adaptée à un groupe d'utilisateurs différent. Publiez les vues métier.
- Créez «Groupes sous forme de schéma en étoile».
- Appliquez la sécurité des objets, si nécessaire.
- Créez des packs et publiez les métadonnées.

Pour en savoir davantage sur les rubriques non traitées ici, reportez-vous au manuel *Framework Manager User Guide*.

### Groupes sous forme de schéma en étoile

Le concept des dimensions conformes n'est pas limité à la modélisation dimensionnelle, il s'applique également aux sujets de requête.

Utilisez l'assistant **Star Schema Grouping** pour créer rapidement des groupes de raccourcis qui vont fournir du contexte à vos utilisateurs sur les appartenances des objets entre eux. Ceci rend le modèle plus intuitif aux yeux des utilisateurs. Les groupes sous forme de schéma en étoile peuvent également faciliter la génération de rapports portant sur des faits multiples en permettant la répétition des dimensions partagées dans différents groupes. Ceci aide les utilisateurs à voir ce que les différents groupes ont en commun et la façon dont ils peuvent générer des rapports inter-fonctionnels ou à faits multiples. Pour en savoir davantage, reportez-vous à la section «Requêtes à faits multiples, requêtes à granularité multiple», à la page 8.

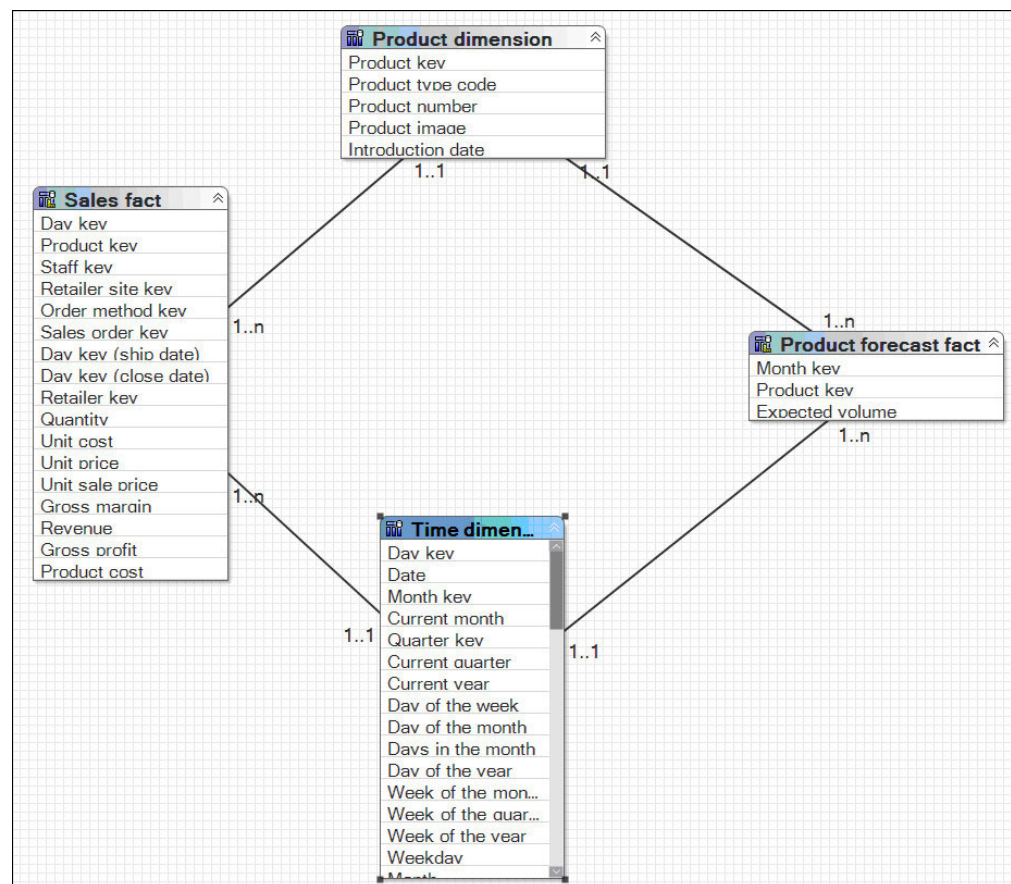
Les groupes sous forme de schéma en étoile fournissent également une aide contextuelle pour les requêtes à plusieurs chemins de jointure. En créant des groupes sous forme de schéma en étoile dans la vue métier du modèle, vous

pouvez déterminer quel chemin de jointure doit être sélectionné lorsqu'il y en a beaucoup. Cette fonctionnalité est particulièrement utile dans le cadre de requêtes sans fait.

### Schémas en étoiles conformes multiples ou requêtes sans faits multiples :

Vous verrez probablement des sujets de requête dimensionnels qui sont joints à plusieurs sujets de requête de faits. L'ambiguïté au niveau des jointures est un problème pouvant se produire lors de la création de rapports à partir d'éléments issus de plusieurs dimensions ou sujets de requête dimensionnels sans inclure aucun élément de la dimension de mesure ou du sujet de requête de fait. On parle alors de requête sans fait.

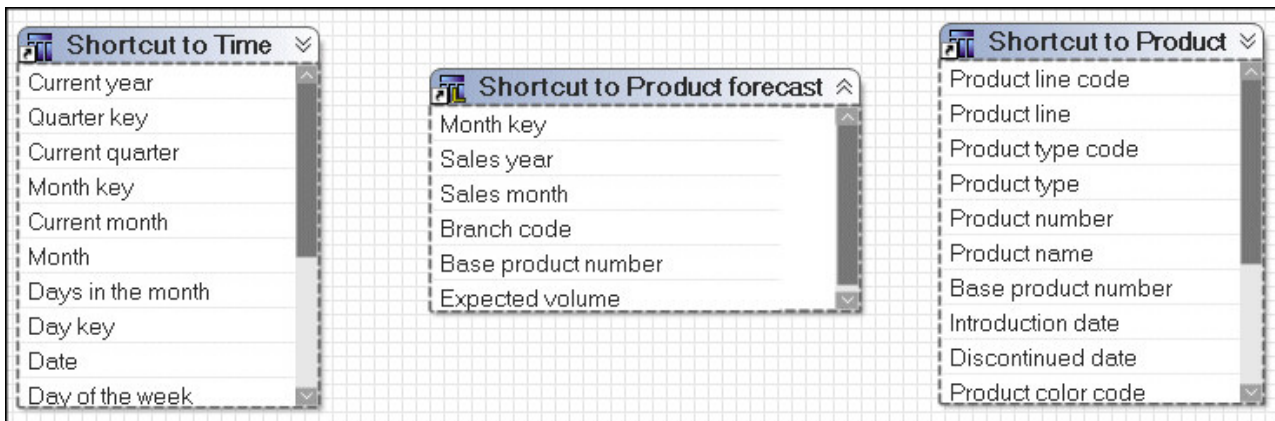
Par exemple, les dimensions Product et Time sont associées aux faits Product forecast et Sales.



A l'aide de ces relations, comment créer un rapport utilisant uniquement des éléments de Product et Year ? La question professionnelle pourrait être de savoir quels produits ont été prévus à la vente en 2005 ou quels produits ont effectivement été vendus en 2005. Même si cette requête implique uniquement Product et Time, ces dimensions sont associées par le biais de plusieurs faits. Il n'existe aucun moyen de savoir quelle question professionnelle est posée. Vous devez définir le contexte pour la requête sans fait.

Dans l'exemple suivant, il est conseillé de créer deux espaces-noms, un contenant des raccourcis vers Product, Time et Product forecast, l'autre contenant Product,

## Time et Sales.



Lorsque vous procédez ainsi pour tous les schémas en étoile, vous résolvez les ambiguïtés de jointure en plaçant les raccourcis pointant vers le fait et toutes les dimensions ordinaires dans un même espace-noms. Les raccourcis associés aux dimensions conformes de chaque espace-noms sont identiques et constituent des références vers l'objet d'origine. **Remarque** : Il s'agit exactement de la même règle que celle appliquée aux dimensions ordinaires et aux dimensions de mesure.

Avec un espace-noms pour chaque schéma en étoile, vos utilisateurs n'ont plus aucun doute concernant les éléments à utiliser. Pour créer un rapport sur les produits réellement vendus en 2005, ils utilisent les éléments Product et Year de l'espace-noms Sales. La seule relation pertinente dans ce contexte est celle associant les éléments Product, Time et Sales ; elle est utilisée pour renvoyer les données.

---

## Chapitre 2. Code SQL généré par IBM Cognos Software

Le code SQL généré par IBM Cognos Software est souvent difficile à interpréter. Le présent document offre une explication du code SQL obtenu lors de situations courantes.

**Remarque :** Les exemples de code SQL indiqués dans le présent document servent à illustrer des cas de figure spécifiques ; leur longueur a été éditée. Ces exemples utilisent la version 8.2 du modèle d'exemple.

Pour accéder au *Guide de modélisation des métadonnées IBM Cognos* dans une autre langue, accédez à `emplacement_installation\c10\webcontent\documentation` et ouvrez le dossier correspondant à la langue choisie. Ouvrez ensuite le fichier `ug_best.pdf`.

---

### Présentation des requêtes dimensionnelles

Les requêtes dimensionnelles sont conçues pour permettre l'utilisation de requêtes à faits multiples.

Les objectifs de base des requêtes à faits multiples sont les suivants :

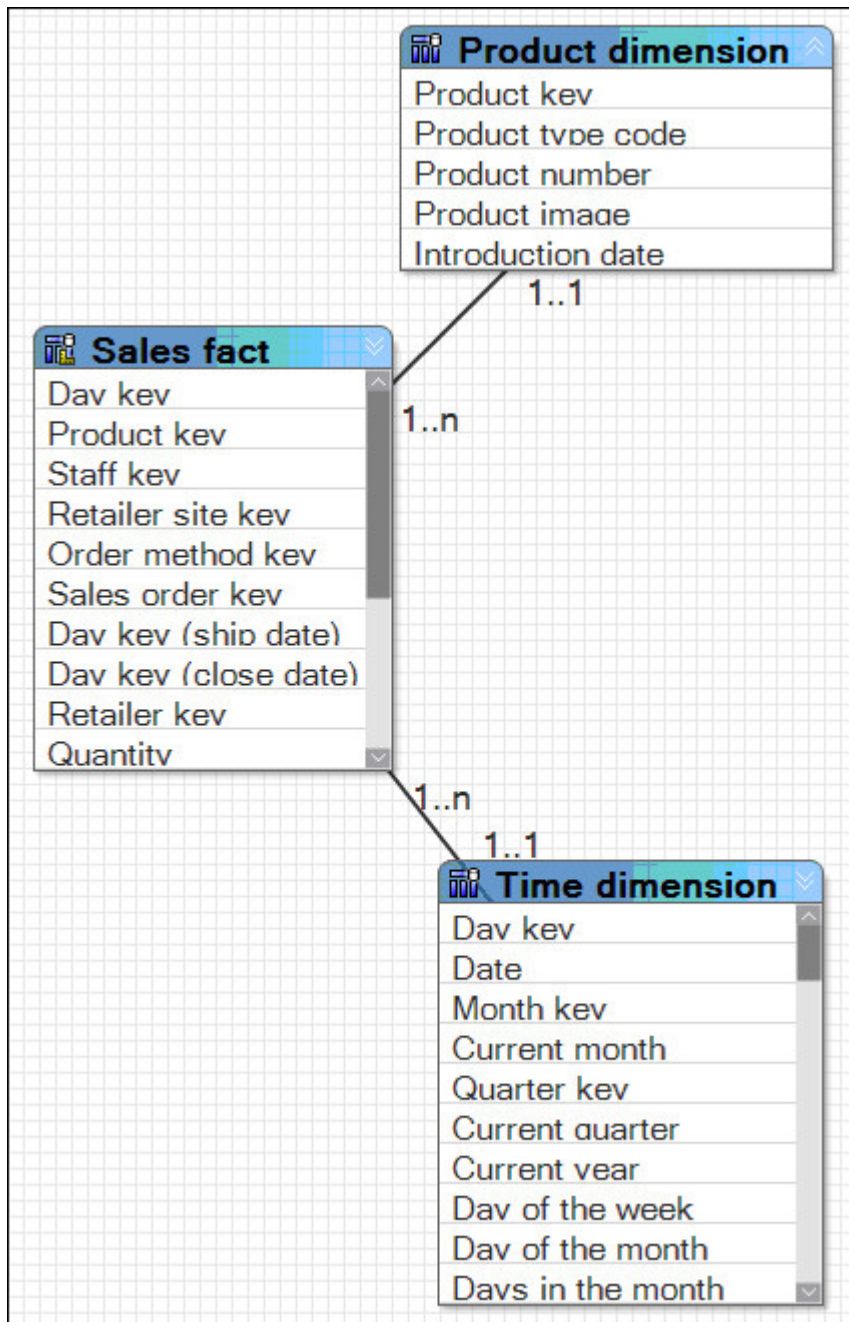
- Conserver les données lorsque les données de faits ne sont pas parfaitement alignées entre deux dimensions communes, par exemple lorsqu'il y a plus de lignes dans les faits que dans les dimensions.
- Eviter le double comptage lorsque des données de faits existent à différents niveaux de granularité, en s'assurant que chaque fait est représenté dans une seule requête avec le type de groupement approprié. Dans certains cas, vous devrez peut-être créer des déterminants pour les sujets de requête sous-jacents.

#### Requête à fait unique

Lorsqu'une requête figure dans un groupe sous forme de schéma en étoile, le résultat obtenu est une requête à fait unique.

Dans cet exemple, toutes les requêtes utilisent le fait Sales. Les dimensions fournissent des attributs et des descriptions qui rendent les données du fait Sales plus explicites. Toutes les relations entre les dimensions et ce fait sont de type 1-n.





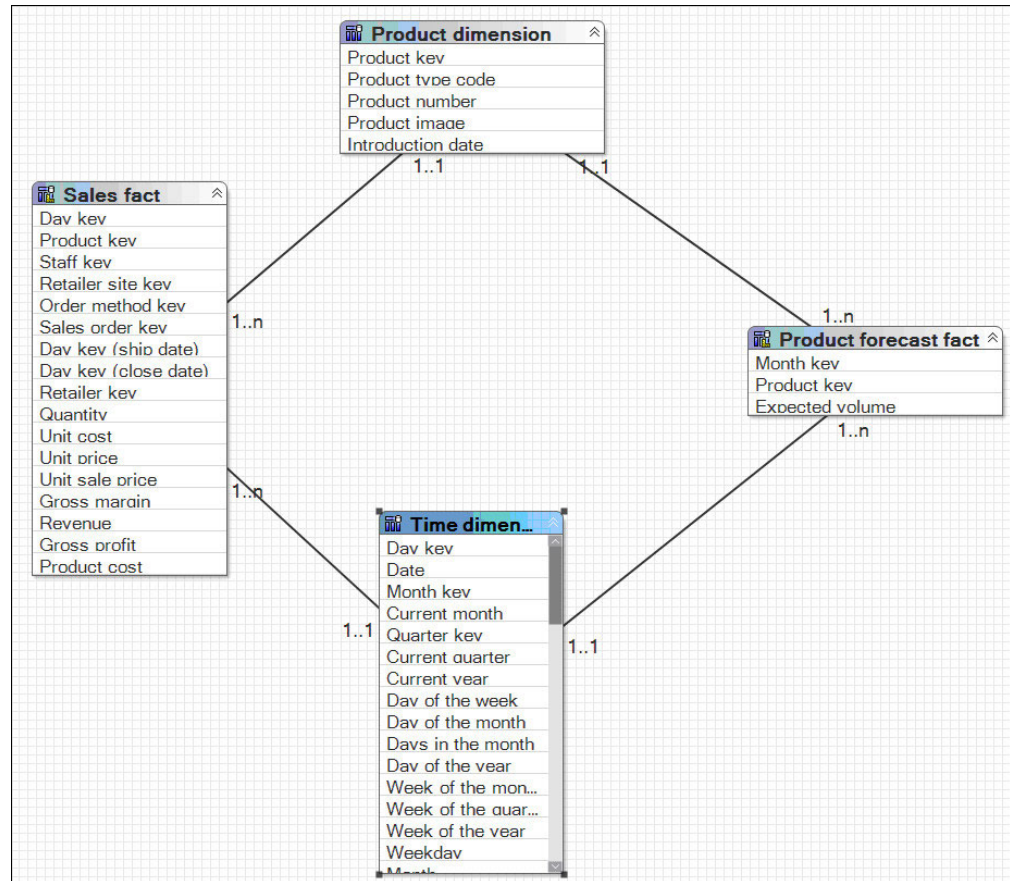
Lorsque vous appliquez un filtre sur Month et Product, vous obtenez le résultat suivant :

MONTH_NAME	PRODUCT_NAME	QUANTITY
April 2004	Aloe Relief	1,410
April 2004	Course Pro Umbrella	132
February 2004	Aloe Relief	270
February 2006	Aloe Relief	88

## Requête à granularité multiple et à faits multiples portant sur des dimensions conformes

Une requête portant sur des faits multiples dans des dimensions conformes respecte la cardinalité entre chaque table de faits et ses dimensions et produit un code SQL pour renvoyer toutes les lignes à partir de chaque table de faits.

Par exemple, Sales et Product Forecast sont tous deux des faits.



Notez qu'il s'agit d'une représentation simplifiée, et non d'un exemple de la façon dont ces données s'affichent dans un modèle créé à partir des recommandations de modélisation d'IBM Cognos.

### Résultat

Des requêtes individuelles portant sur les faits Sales, Product Forecast en fonction de Month et Product donnent les résultats suivants. Les données du fait Sales sont stockées au niveau des jours.



MONTH_NAME	PRODUCT_NAME	EXPECTED_VOLUME
April 2004	Aloe Relief	1,690
April 2004	Course Pro Umbrella	125
February 2004	Aloe Relief	246
February 2004	Course Pro Umbrella	1
February 2006	Aloe Relief	92

Une requête portant sur les faits Sales et Product Forecast respecte la cardinalité entre chaque table de faits et ses dimensions et produit un code SQL pour renvoyer toutes les lignes à partir de chaque table de faits. Les tables de faits sont mises en correspondance en fonction de leurs clés communes, à savoir Month et Product, et sont agrégées au niveau de granularité commun le plus bas dès que cela est possible. Dans ce cas, les jours sont cumulés pour constituer des mois. Des valeurs nulles sont souvent renvoyées pour ce type de requête car il est fort possible qu'une combinaison d'éléments dimensionnels figurant dans une table de faits n'existe pas dans l'autre table.

MONTH_NAME	PRODUCT_NAME	QUANTITY	EXPECTED_VOLUME
April 2004	Aloe Relief	1,410	1,690
April 2004	Course Pro Umbrella	132	125
February 2004	Aloe Relief	270	246
February 2004	Course Pro Umbrella		1
February 2006	Aloe Relief	88	92

Notez qu'en février 2004, l'élément Course Pro Umbrella faisait partie des prévisions mais aucun article de cette ligne n'a été vendu. Les données des faits Sales et Product Forecast existent à différents niveaux de granularité. Les données du fait Sales sont associées au niveau des jours, celles du fait Product Forecast au niveau des mois.

## Code SQL

Le code SQL généré par le logiciel IBM Cognos également appelé requête liée, est souvent difficile à interpréter. Une requête liée utilise plusieurs sous-requêtes, une pour chaque étoile, rassemblées par une jointure externe intégrale reliant les clés communes. L'objectif est de conserver tous les membres dimensionnels se produisant de chaque côté de la requête.

L'exemple suivant, raccourci pour des raisons pratiques, permet d'illustrer les fonctions principales des requêtes liées.

```
select
  coalesce(D2.MONTH_NAME,D3.MONTH_NAME) as MONTH_NAME,
  coalesce(D2.PRODUCT_NAME,D3.PRODUCT_NAME) as PRODUCT_NAME,
  D2.EXPECTED_VOLUME as EXPECTED_VOLUME,
  D3.QUANTITY as QUANTITY
from (select TIME.MONTH_NAME as MONTH_NAME,
  PRODUCT_LOOKUP.PRODUCT_NAME as PRODUCT_NAME,
  XSUM(PRODUCT_FORECAST_FACT.EXPECTED_VOLUME for
  TIME.CURRENT_YEAR,TIME.QUARTER_KEY,TIME.MONTH_KEY,
  PRODUCT.PRODUCT_LINE_CODE, PRODUCT.PRODUCT_TYPE_CODE,
  PRODUCT.PRODUCT_KEY) as EXPECTED_VOLUME
```

```

from
  (select TIME.CURRENT_YEAR as CURRENT_YEAR,
    TIME.QUARTER_KEY as QUARTER_KEY,
    TIME.MONTH_KEY as MONTH_KEY,
    XMIN(TIME.MONTH_NAME for TIME.CURRENT_YEAR,
    TIME.QUARTER_KEY,TIME.MONTH_KEY) as MONTH_NAME
  from TIME_DIMENSION TIME
  group by TIME.MONTH_KEY) TIME
join PRODUCT_FORECAST_FACT PRODUCT_FORECAST_FACT
on (TIME.MONTH_KEY = PRODUCT_FORECAST_FACT.MONTH_KEY)
join PRODUCT_PRODUCT on (PRODUCT.PRODUCT_KEY =
PRODUCT_FORECAST_FACT.PRODUCT_KEY)
where
  (PRODUCT.PRODUCT_NAME in ('Aloe Relief','Course Pro
Umbrella')) and
  (TIME.MONTH_NAME in ('April 2004','February 2004','February
2006'))
group by
  TIME.MONTH_NAME,
  PRODUCT_LOOKUP.PRODUCT_NAME
) D2
full outer join
(select TIME.MONTH_NAME as MONTH_NAME,
  PRODUCT_LOOKUP.PRODUCT_NAME as PRODUCT_NAME,
  XSUM(SALES_FACT.QUANTITY for TIME.CURRENT_YEAR,
  TIME.QUARTER_KEY, TIME.MONTH_KEY,
  PRODUCT.PRODUCT_LINE_CODE, PRODUCT.PRODUCT_TYPE_CODE,
  PRODUCT.PRODUCT_KEY ) as QUANTITY
from
  select TIME.DAY_KEY,TIME.MONTH_KEY,TIME.QUARTER_KEY,
  TIME.CURRENT_YEAR,TIME.MONTH_EN as MONTH_NAME
  from TIME_DIMENSION TIME) TIME
join SALES_FACT SALES_FACT
on (TIME.DAY_KEY = SALES_FACT.ORDER_DAY_KEY)
join PRODUCT_PRODUCT on (PRODUCT.PRODUCT_KEY = SALES_FACT.PRODUCT_KEY)
where
  PRODUCT.PRODUCT_NAME in ('Aloe Relief','Course Pro Umbrella'))
  and (TIME.MONTH_NAME in ('April 2004','February 2004','February
2006'))
group by
  TIME.MONTH_NAME,
  PRODUCT.PRODUCT_NAME
) D3
on ((D2.MONTH_NAME = D3.MONTH_NAME) and
(D2.PRODUCT_NAME = D3.PRODUCT_NAME))

```

## Qu'est-ce qu'une instruction Coalesce ?

Une instruction coalesce est simplement un procédé efficace permettant de traiter les éléments de requêtes à partir de dimensions conformes. Elle est utilisée pour accepter la première valeur non nulle retournée par un des sujets de requête. Cette instruction accepte une liste exhaustive de clés sans répétition lors de l'application d'une jointure externe intégrale.

## A quoi sert une jointure externe intégrale ?

Une jointure externe intégrale est nécessaire pour garantir que toutes les données de chaque table de faits sont extraites. Une jointure interne ne donne des résultats que si un article en stock a été vendu. Une jointure externe droite renvoie toutes les ventes relatives aux articles en stock. Une jointure externe gauche renvoie tous les articles en stock ayant fait l'objet d'une vente. Une jointure externe intégrale est le seul moyen de savoir ce qui était en stock et ce qui a été vendu.

## Modélisation de relations de type 1-n en relations de type 1-1

Si une relation de type 1-n existe dans les données, mais est modélisée en tant que relation de type 1-1, les interruptions SQL sont inévitables, car les informations fournies au logiciel IBM Cognos par les métadonnées ne sont pas suffisantes.

Les problèmes les plus courants qui se produisent lorsque des relations de type 1-n sont modélisées en tant que relations de type 1-1 sont les suivants :

- Le système n'empêche pas automatiquement le double comptage des requêtes à granularité multiple.

Le logiciel IBM Cognos ne parvient pas à détecter les faits et génère donc une requête liée pour compenser le double comptage qui peut se produire dans le cas de relations hiérarchiques et de différents niveaux de granularité au sein des dimensions conformes.

- Les requêtes à faits multiples ne sont pas automatiquement détectées.

Le logiciel IBM Cognos ne dispose pas de suffisamment d'informations pour détecter une requête à faits multiples. Pour les requêtes à faits multiples, une jointure interne est appliquée et la jointure de boucle est éliminée suite à la suppression de la dernière jointure évaluée. Le fait de supprimer une jointure peut entraîner des résultats imprévisibles, voire incorrects, en fonction des dimensions et des faits inclus dans la requête.

Si la cardinalité était modifiée de façon à n'utiliser que des relations de type 1-1 entre les sujets de requête ou les dimensions, le résultat d'une requête portant sur les faits Product Forecast et Sales avec Time ou Time et Product générerait une instruction Select dont le rôle est de supprimer une jointure pour éviter une référence circulaire.

L'exemple ci-dessous montre que les résultats de cette requête sont incorrects si on les compare à ceux de requêtes individuelles portant sur les faits Sales ou Product Forecast.

Voici les résultats obtenus pour des requêtes individuelles :

MONTH_NAME	PRODUCT_NAME	QUANTITY
April 2004	Aloe Relief	1,410
April 2004	Course Pro Umbrella	132
February 2004	Aloe Relief	270
February 2006	Aloe Relief	88

MONTH_NAME	PRODUCT_NAME	EXPECTED_VOLUME
April 2004	Aloe Relief	1,690
April 2004	Course Pro Umbrella	125
February 2004	Aloe Relief	246
February 2004	Course Pro Umbrella	1
February 2006	Aloe Relief	92

Lorsque vous associez ces requêtes pour n'en faire qu'une seule, les résultats sont les suivants :

MONTH_NAME	PRODUCT_NAME	QUANTITY	EXPECTED_VOLUME
April 2004	Aloe Relief	68,598	1,811,680
April 2004	Course Pro Umbrella	68,598	134,000
February 2004	Aloe Relief	29,672	105,780
February 2004	Course Pro Umbrella	29,672	430
February 2006	Aloe Relief	28,564	47,196

## Code SQL

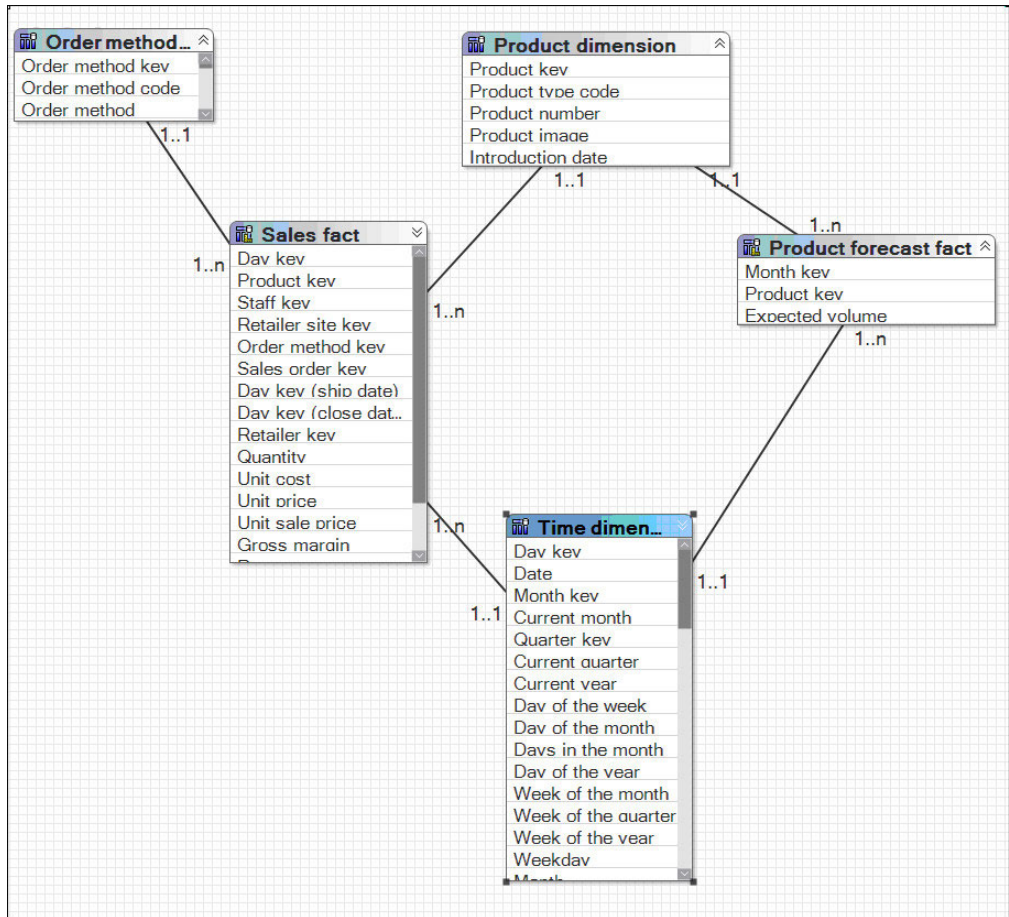
En raison d'un chemin de jointure circulaire détecté dans le modèle, le code SQL généré n'a pas inclus l'une des relations qui n'étaient pas nécessaires pour compléter le chemin de jointure. Dans l'exemple suivant, la relation entre les faits Time et Product Forecast a été supprimée.

Un chemin d'accès circulaire produit rarement une requête donnant des résultats utiles.

```
select
  TIME.MONTH_NAME as MONTH_NAME,
  PRODUCT_LOOKUP.PRODUCT_NAME as PRODUCT_NAME,
  XSUM(SALES_FACT.QUANTITY for
  TIME.CURRENT_YEAR, TIME.QUARTER_KEY, TIME.MONTH_KEY,
  PRODUCT.PRODUCT_LINE_CODE, PRODUCT.PRODUCT_TYPE_CODE,
  PRODUCT.PRODUCT_KEY ) as QUANTITY,
  XSUM(PRODUCT_FORECAST_FACT.EXPECTED_VOLUME for TIME.CURRENT_YEAR,
  TIME.QUARTER_KEY, TIME.MONTH_KEY, PRODUCT.PRODUCT_LINE_CODE,
  PRODUCT.PRODUCT_TYPE_CODE, PRODUCT.PRODUCT_KEY ) as EXPECTED_VOLUME
from
  (select TIME.DAY_KEY,TIME.MONTH_KEY, TIME.QUARTER_KEY,
  TIME.CURRENT_YEAR,TIME.MONTH_EN as MONTH_NAME
  from TIME_DIMENSION TIME) TIME
join
  SALES_FACT on (TIME.DAY_KEY = SALES_FACT.ORDER_DAY_KEY)
join
  PRODUCT_FORECAST_FACT on (TIME.MONTH_KEY =
  PRODUCT_FORECAST_FACT.MONTH_KEY)
join
  PRODUCT (PRODUCT.PRODUCT_KEY = PRODUCT_FORECAST_FACT.PRODUCT_KEY)
where
  (PRODUCT.PRODUCT_NAME in ('Aloe Relief','Course Pro Umbrella'))
and
  (TIME.MONTH_NAME in ('April 2004','February 2004','February 2006'))
group by
  TIME.MONTH_NAME, PRODUCT.PRODUCT_NAME
```

## Requête à granularité multiple et à faits multiples portant sur des dimensions non conformes

Si une dimension non conforme est ajoutée à la requête, la nature du résultat produit par la requête liée change. Les enregistrements ne peuvent plus être agrégés au niveau de granularité commun le plus bas, puisqu'un côté de la requête présente un caractère dimensionnel non commun à l'autre côté. Le résultat retourné est formé de deux listes en corrélation.



## Résultat

Voici à quoi ressemblent les résultats de requêtes individuelles portant sur leur schéma en étoile respectif :

MONTH_NAME	PRODUCT_NAME	ORDER_METHOD	QUANTITY
April 2004	Aloe Relief	E-mail	114
April 2004	Aloe Relief	Fax	220
April 2004	Aloe Relief	Mail	100
April 2004	Aloe Relief	Sales visit	322
April 2004	Aloe Relief	Telephone	286
April 2004	Aloe Relief	Web	368
April 2004	Course Pro Umbrella	E-mail	22
April 2004	Course Pro Umbrella	Fax	28
April 2004	Course Pro Umbrella	Sales visit	82
February 2004	Aloe Relief	Mail	28
February 2004	Aloe Relief	Sales visit	102
February 2004	Aloe Relief	Telephone	28
February 2004	Aloe Relief	Web	112
February 2006	Aloe Relief	Sales visit	88

MONTH_NAME	PRODUCT_NAME	QUANTITY	EXPECTED_VOLUME
April 2004	Aloe Relief	1,410	1,690
April 2004	Course Pro Umbrella	132	125
February 2004	Aloe Relief	270	246
February 2004	Course Pro Umbrella		1
February 2006	Aloe Relief	88	92

Une requête portant sur les mêmes éléments des deux schémas en étoile produit le résultat suivant.



MONTH_NAME	PRODUCT_NAME	ORDER_METHOD	QUANTITY	EXPECTED_VOLUME
April 2004	Aloe Relief	Sales visit	322	1,690
April 2004	Aloe Relief	Telephone	286	1,690
April 2004	Aloe Relief	Web	368	1,690
April 2004	Aloe Relief	E-mail	114	1,690
April 2004	Aloe Relief	Fax	220	1,690
April 2004	Aloe Relief	Mail	100	1,690
April 2004	Course Pro Umbrella	E-mail	22	125
April 2004	Course Pro Umbrella	Fax	28	125
April 2004	Course Pro Umbrella	Sales visit	82	125
February 2004	Aloe Relief	Web	112	246
February 2004	Aloe Relief	Mail	28	246
February 2004	Aloe Relief	Sales visit	102	246
February 2004	Aloe Relief	Telephone	28	246
February 2004	Course Pro Umbrella			1
February 2006	Aloe Relief	Sales visit	88	92

Dans ce résultat, le niveau de granularité le plus bas correspondant aux enregistrements du fait Sales renvoie un nombre plus important d'enregistrements pour chaque combinaison Month/Product. Il existe maintenant une relation de type 1-n entre les lignes renvoyées par le fait Product Forecast et celles renvoyées par le fait Sales.

Lorsque vous comparez cette situation au résultat obtenu dans l'exemple de requête à granularité multiple et à faits multiples portant sur des dimensions conformes, vous pouvez constater que davantage d'enregistrements sont renvoyés et que les résultats relatifs au fait Expected Volume sont répétés dans plusieurs faits Order Method. Si vous ajoutez le fait Order Method à la requête, la relation entre les données Quantity et Expected Volume se transforme en relation de type 1-n. Il n'est alors plus possible de relier une valeur unique du fait Expected Volume à une valeur du fait Quantity.

Le fait d'appliquer un groupement à la clé Month démontre que le résultat de cet exemple repose sur le même jeu de données que celui de la requête à granularité multiple et à faits multiples, mais avec un degré de granularité plus élevé.

## Code SQL

Le code SQL lié créé pour cet exemple est très semblable au code SQL créé dans la requête à granularité multiple et à faits multiples. La différence principale réside dans l'ajout du fait Order Method. Ce dernier ne constitue pas une dimension conforme et n'a une incidence que sur la requête portant sur la table du fait Sales.

```
select
  D2.QUANTITY as QUANTITY,
  D3.EXPECTED_VOLUME as EXPECTED_VOLUME,
  coalesce(D2.PRODUCT_NAME,D3.PRODUCT_NAME) as PRODUCT_NAME,
  coalesce(D2.MONTH_NAME,D3.MONTH_NAME) as MONTH_NAME,
  D2.ORDER_METHOD as ORDER_METHOD
from
  (select
    PRODUCT.PRODUCT_NAME as PRODUCT_NAME,
    TIME.MONTH_NAME as MONTH_NAME,
    ORDER_METHOD.ORDER_METHOD as ORDER_METHOD,
```



```

XSUM(SALES_FACT.QUANTITY for TIME.CURRENT_YEAR,TIME.QUARTER_KEY,
TIME.MONTH_KEY,PRODUCT.PRODUCT_LINE_CODE,PRODUCT.PRODUCT_TYPE_CODE,
PRODUCT.PRODUCT_KEY,ORDER_METHOD_DIMENSION.ORDER_METHOD_KEY) as
QUANTITY
from
PRODUCT_DIMENSION PRODUCT
join
SALES_FACT SALES_FACT
on (PRODUCT.PRODUCT_KEY = SALES_FACT.PRODUCT_KEY)
join
ORDER_METHOD_DIMENSION ORDER_METHOD
on (ORDER_METHOD.ORDER_METHOD_KEY = SALES_FACT.ORDER_METHOD_KEY)
join TIME_DIMENSION TIME
on ( TIME.DAY_KEY = SALES_FACT.ORDER_DAY_KEY)
where
(PRODUCT.PRODUCT_NAME in ('Aloe Relief','Course Pro Umbrella'))
and
( TIME.MONTH_NAME in ('April 2004','February 2004','February 2006'))
group by
PRODUCT.PRODUCT_NAME,
TIME.MONTH_NAME,
ORDER_METHOD.ORDER_METHOD
) D2
full outer join
(select
PRODUCT.PRODUCT_NAME as PRODUCT_NAME,
TIME.MONTH_NAME as MONTH_NAME,
XSUM(PRODUCT_FORECAST_FACT.EXPECTED_VOLUME for TIME.CURRENT_YEAR,
TIME.QUARTER_KEY,TIME.MONTH_KEY,PRODUCT.PRODUCT_LINE_CODE,
PRODUCT.PRODUCT_TYPE_CODE,PRODUCT.PRODUCT_KEY) as EXPECTED_VOLUME
from
PRODUCT_DIMENSION PRODUCT
join
PRODUCT_FORECAST_FACT PRODUCT_FORECAST_FACT
on (PRODUCT.PRODUCT_KEY = PRODUCT_FORECAST_FACT.PRODUCT_KEY)
join
(select
TIME.CURRENT_YEAR as CURRENT_YEAR,
TIME.QUARTER_KEY as QUARTER_KEY,
TIME.MONTH_KEY as MONTH_KEY,
XMIN(TIME.MONTH_NAME for TIME.CURRENT_YEAR, TIME.QUARTER_KEY,
TIME.MONTH_KEY) as MONTH_NAME
from
TIME_DIMENSION TIME
group by
TIME.CURRENT_YEAR,
TIME.QUARTER_KEY,
TIME.MONTH_KEY
) TIME
on (TIME.MONTH_KEY = PRODUCT_FORECAST_FACT.MONTH_KEY)
where
(PRODUCT.PRODUCT_NAME in ('Aloe Relief','Course Pro Umbrella'))
and
(TIME.MONTH_NAME in ('April 2004','February 2004','February 2006'))
group by
PRODUCT.PRODUCT_NAME,
TIME.MONTH_NAME
) D3
on ((D2.PRODUCT_NAME = D3.PRODUCT_NAME) and
(D2.MONTH_NAME = D3.MONTH_NAME))

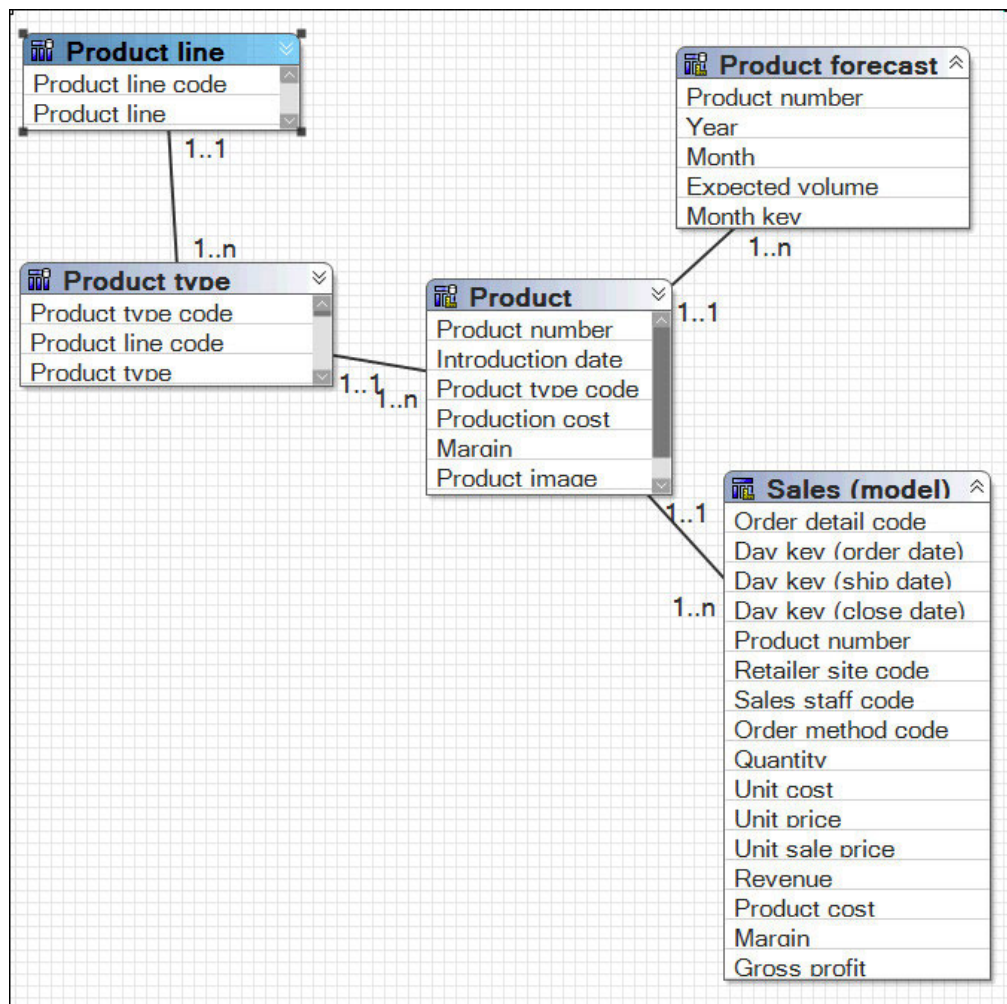
```

## Résolutions des dimensions et faits identifiés de façon ambiguë

Un sujet de requête est considéré comme étant défini de façon ambiguë s'il participe à la fois à la relation n et à la relation 1 avec d'autres sujets de requête. Un sujet de requête défini de façon ambiguë n'est pas nécessairement dangereux du point de vue de la création de requêtes. Nous vous conseillons d'utiliser les cas présentés ci-dessous pour évaluer les sujets de requête. Le but de cette évaluation est d'éviter tout fractionnement de requête non nécessaire et de vérifier que les fractionnements qui se produisent sont intentionnels et corrects.

### Sujets de requête représentant un niveau de hiérarchie

Un cas fréquent de sujet de requête défini de façon ambiguë qui n'est pas dangereux est celui où le sujet de requête représente un niveau intermédiaire d'une hiérarchie descriptive. Un exemple en est la Hiérarchie Produits qui suit.



Dans cet exemple, les dimensions Product type et Product pourraient être identifiées comme étant définies de façon ambiguë, mais cette ambiguïté n'est négative ni pour les résultats produits, ni pour les performances des requêtes faisant appel à un ou plusieurs de ces sujets de requête. Vous n'avez pas besoin de corriger ce modèle de requête car d'après les règles de détection de fait, un seul fait est identifié dans n'importe quelle requête qui associe un élément des sujets de

requête Product forecast ou Sales. Il est recommandé de réduire les hiérarchies en une seule dimension ordinaire lors d'une modélisation à des fins d'analyse.

Exemples de requêtes pouvant être écrites à l'aide de cet exemple :

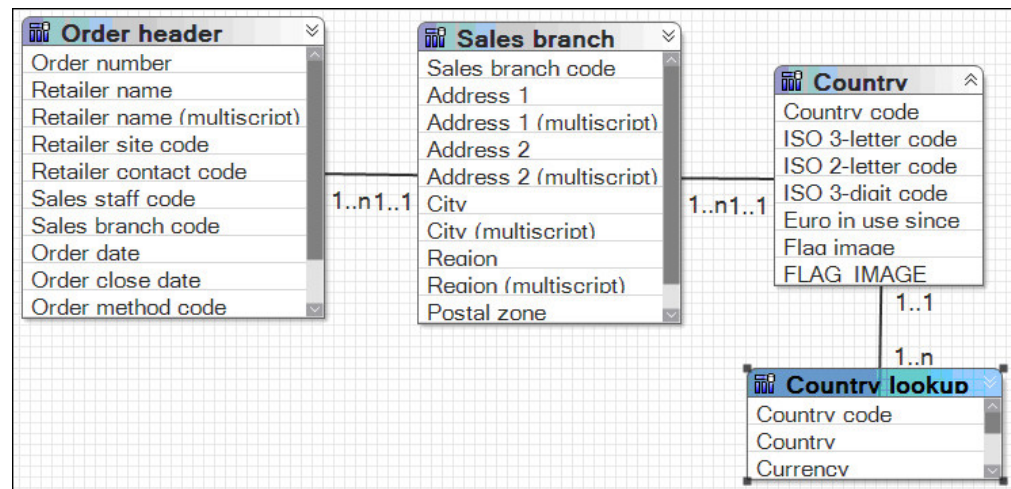
Les éléments des sujets de requête ci-dessous sont utilisés dans une requête :	Sujet de requête se comportant comme un fait dans la requête :
Product line et Product type	Type de produit
Product line, Product type et Product	Produit
Product line, Product type, Product et Sales	Sales
Product line et Sales	Sales
Product type et Product forecast	Product forecast

## Résolution de requêtes fractionnées par erreur

Si des requêtes sont fractionnées alors qu'elles n'auraient pas dû l'être, vous devez les résoudre.

Les sujets de requêtes situés du côté n de toutes les relations sont identifiés comme étant des faits. Dans l'exemple qui suit, vous constatez que les sujets Order Header et Country Multilingual se comportent comme des faits. En réalité, le sujet de requête Country Multilingual ne contient que des informations descriptives et sert vraisemblablement de table de recherche. Selon une perspective de dimension ou de modélisation professionnelle, le sujet de requête Country Multilingual est une extension du sujet Country.

En quoi est-ce un problème de laisser le modèle tel quel ?



Testez ce modèle en créant un rapport relatif au nombre de commandes par ville puis par pays ou région. Lorsque vous utilisez ce modèle, vous obtenez un résultat incorrect. Les chiffres sont exacts pour les villes mais certaines villes ne sont pas associées au bon pays ou région. Voici un exemple de résultat lié de façon incorrecte.

COUNTRY	CITY	Number of Orders
Australia	Amsterdam	279
Austria	Bilbao	123
Belgium	Birmingham	164
Brazil	Boston	515
Canada	Calgary	123

Habituellement, il s'agit du point de départ lorsque vous voyez ce genre de résultat dans le code SQL.

## Code SQL

Dans cet exemple, vous voyez une requête liée, ce qui est logique si le modèle contient des faits multiples. Par définition, une requête liée est une requête qui tente de lier plusieurs faits entre eux. Elle utilise les relations qui existent entre les faits, ainsi que les déterminants des dimensions conformes (ou communes) définies dans le modèle. Une requête liée peut être identifiée par deux requêtes disposant d'une jointure externe intégrale. Une requête d'encapsuleur doit inclure une instruction coalesce portant sur les dimensions conformes.

Notez les problèmes suivants dans le code SQL :

- La requête ne dispose d'aucune instruction coalesce.
- RSUM indique une tentative de création d'une clé valide..

```

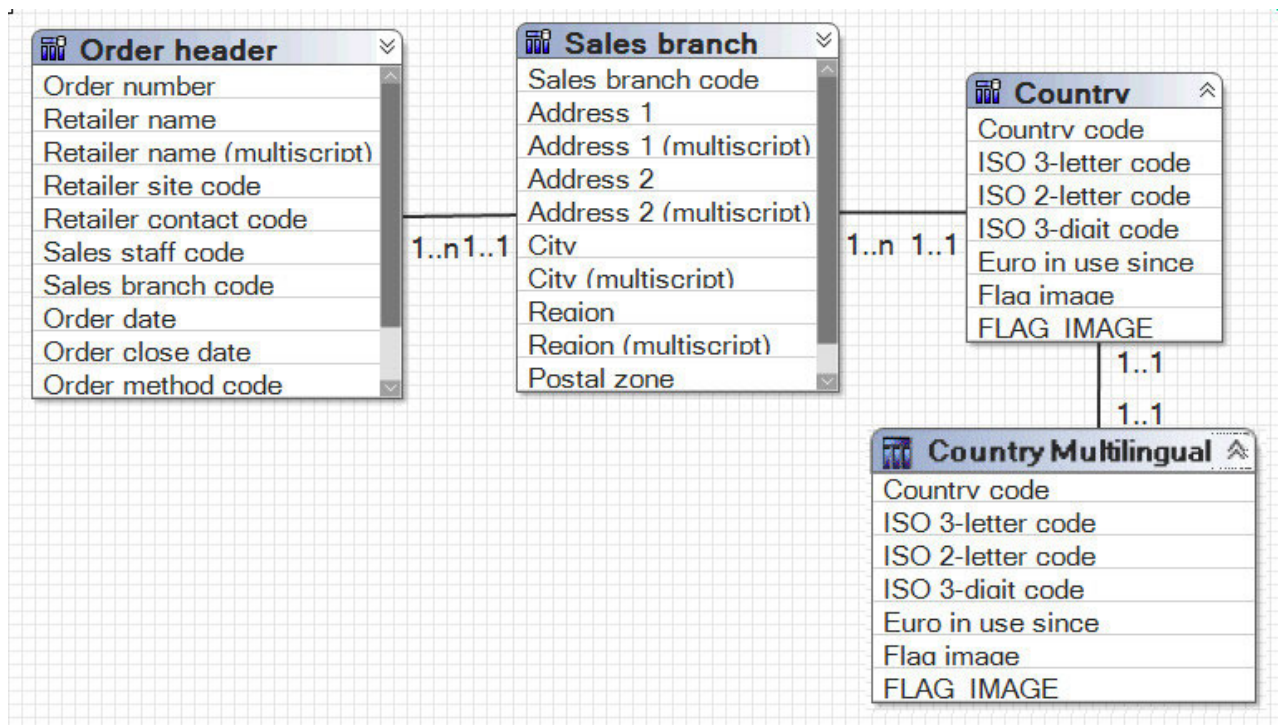
select
  D3.COUNTRY as COUNTRY,
  D2.CITY as CITY,
  D2.number_of_orders as number_of_orders
from
  (select
    SALES_BRANCH.CITY as CITY,
    XCOUNT(ORDER_HEADER.ORDER_NUMBER for SALES_BRANCH.CITY) as
    number_of_orders,
    RSUM(1 at SALES_BRANCH.CITY order by SALES_BRANCH.CITY
    asc local)
    as sc
  from
    gosales.gosales.dbo.SALES_BRANCH SALES_BRANCH
  join
    gosales.gosales.dbo.ORDER_HEADER ORDER_HEADER
  on (SALES_BRANCH.SALES_BRANCH_CODE = ORDER_HEADER.SALES_BRANCH_CODE)
  group by
    SALES_BRANCH.CITY
  order by
    CITY asc
  ) D2
full outer join
  (select
    COUNTRY_MULTILINGUAL.COUNTRY as COUNTRY,
    RSUM(1 at COUNTRY_MULTILINGUAL.COUNTRY order by
    COUNTRY_MULTILINGUAL.COUNTRY asc local) as sc
  from
    gosales.gosales.dbo.COUNTRY_MULTILINGUAL COUNTRY_MULTILINGUAL
  group by
    COUNTRY_MULTILINGUAL.COUNTRY
  order by
    COUNTRY asc
  ) D3
on (D2.sc = D3.sc)

```

En examinant les colonnes liées de chaque requête, vous constatez qu'elles sont calculées selon des critères non connexes. Cela explique pourquoi il n'existe apparemment aucune relation entre les pays ou les régions et les villes cités dans le rapport.

Pourquoi une requête liée s'affiche-t-elle ? Pour répondre à cette question, vous devez examiner le modèle.

Dans cet exemple, les éléments de requête utilisés dans le rapport provenaient de différents sujets de requête. Country ou region provenait de Country Multilingual, City de Sales Branch et Number of Orders d'un nombre issu de Order Number dans le sujet de requête Order Header.



Le problème vient du fait que la requête est fractionnée parce que le moteur de requêtes la considère comme une requête à faits multiples. Toutefois, le fractionnement n'est pas associé à une clé valide à partir de laquelle établir une liaison, puisque aucun élément n'est commun aux deux faits.

Pour résoudre ce problème, vous disposez de deux méthodes, requérant toutes les deux une bonne compréhension des données.

### Solution 1

Vous pouvez ajouter un filtre au sujet Country Multilingual pour changer la cardinalité de la relation en 1-1.

```
Select *
from [GOSL].COUNTRY_MULTILINGUAL
Where
COUNTRY_MULTILINGUAL."LANGUAGE"='EN'
```

Vous pouvez également ajouter un filtre portant sur la relation et remplacer la cardinalité par 1-1.

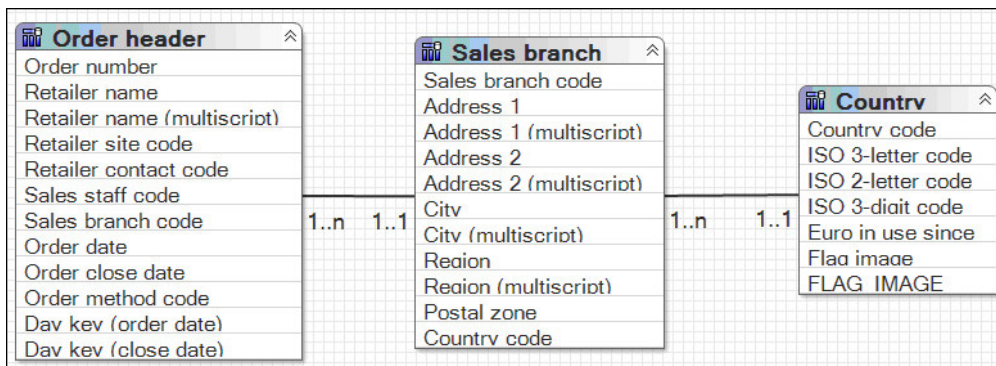


COUNTRY.COUNTRY\_CODE = COUNTRY\_MULTILINGUAL.COUNTRY\_CODE  
 and COUNTRY\_MULTILINGUAL.LANGUAGE = 'EN'

Ces deux possibilités entraînent la création d'un modèle possédant un fait unique pour cette requête.

## Solution 2

Simplifiez le modèle en regroupant les sujets de requête connexes. Cette solution est très avantageuse, puisque le modèle est simplifié et les possibilités d'erreur liées à la création de requêtes sont réduites.



Que vous optiez pour l'une ou l'autre de ces solutions, le résultat de la requête est maintenant correct.

COUNTRY	CITY	Number of Orders
Australia	Melbourne	98
Austria	Wien	162
Belgium	Heverlee	94
Brazil	São Paulo	15
Canada	Calgary	123
Canada	Toronto	330

Le code SQL ne correspond plus à une requête liée.

```
select
  Country.c7 as COUNTRY,
  SALES_BRANCH.CITY as CITY,
  XCOUNT(ORDER_HEADER.ORDER_NUMBER for Country.c7,SALES_BRANCH.CITY)
  as number_of_orders
from
  (select
    COUNTRY.COUNTRY_CODE as c1,
    COUNTRY_MULTILINGUAL.COUNTRY as c7
  from
    gosales.gosales.dbo.COUNTRY COUNTRY
  join
    gosales.gosales.dbo.COUNTRY_MULTILINGUAL COUNTRY_MULTILINGUAL
  on (COUNTRY.COUNTRY_CODE = COUNTRY_MULTILINGUAL.COUNTRY_CODE)
  where COUNTRY_MULTILINGUAL.LANGUAGE='EN'
  ) Country
join
```



```
gosales.gosales.dbo.SALES_BRANCH SALES_BRANCH
on (SALES_BRANCH.COUNTRY_CODE = Country.c1)
join
gosales.gosales.dbo.ORDER_HEADER ORDER_HEADER
on (SALES_BRANCH.SALES_BRANCH_CODE = ORDER_HEADER.SALES_BRANCH_CODE)
group by
Country.c7,
SALES_BRANCH.CITY
```



---

## Remarques

Le présent document a été développé pour des produits et des services proposés dans de nombreux pays.

Le présent document peut être mis à disposition par IBM dans d'autres langues. Toutefois, une copie du produit ou de la version du produit dans cette langue peut être nécessaire pour y accéder.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM. Le présent document peut décrire des produits, des services ou des fonctions qui ne sont pas inclus dans le Logiciel ni dans l'autorisation d'utilisation que vous avez acquise.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations  
IBM Canada Ltd  
3600 Steeles Avenue East  
Markham, Ontario  
L3R 9Z7  
Canada

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit auprès d' IBM à l'adresse suivante :

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japan

Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Software Group  
Attention: Licensing  
3755 Riverside Dr.  
Ottawa, ON K1V 1B7  
Canada

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions de l'ICA, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer

l'exactitude de leurs performances ni leur compatibilité. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

Si vous visualisez ces informations en ligne, il se peut que les photographies et illustrations en couleur n'apparaissent pas à l'écran.

Selon la configuration déployée, la présente Offre Logiciels peut utiliser des cookies de session et des cookies persistants destinés à collecter

- le nom
- le nom d'utilisateur
- le mot de passe

de l'utilisateur pour les fonctions

- de gestion de session
- d'authentification
- de facilité d'utilisation des produits
- de configuration de la connexion unique
- de suivi de l'utilisation, ou pour des fonctions autres que celles-ci.

Ces cookies ne peuvent pas être désactivés.

Si les configurations déployées pour cette offre logicielle vous permettent en tant que client de collecter des informations identifiant la personne provenant des utilisateurs finaux via des cookies et autres technologies, il est conseillé de vous procurer un avis juridique à propos des lois applicables à une telle collecte de données, notamment des exigences au sujet de l'avis et du consentement.

Pour plus d'informations à propos de l'utilisations de nombreuses technologies, y compris de celle des cookies, voir les règles de confidentialité IBM dans <http://www.ibm.com/privacy> ainsi que la politique sur la protection des renseignements personnels IBM en ligne dans <http://www.ibm.com/privacy/details> dans la section appelée "Cookies, balises Web et autres technologies" et les "Règles de confidentialité de produits logiciels et logiciels sous forme de services IBM" dans <http://www.ibm.com/software/info/product-privacy>.

---

## Marques

IBM, le logo IBM et [ibm.com](http://www.ibm.com) sont des marques d'International Business Machines Corp. dans de nombreux pays. Les autres noms de produits et de services peuvent appartenir à IBM ou à d'autres sociétés. La liste actualisée de toutes les marques d'IBM est disponible sur la page Web «Copyright and trademark information» à [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).





---

# Index

## A

agrégation 4  
agrégation de calculs 17  
ambiguïté, relations 23

## B

boucles, jointures 22, 26

## C

calculs,  
  Ordre des opérations 17  
cardinalité  
  1-n 42  
  vérification 22  
cardinalité facultative 2  
cardinalité maximale 2  
cardinalité minimale 2  
cardinality  
  1-1 42  
  dimensions et faits 22  
  règles 2  
  requêtes 2  
  types 2  
concepts 1  
création  
  dimensions de mesure 33  
  groupes sous forme de schémas en étoile 34  
  ordinaires, dimensions 31

## D

déterminants,  
  définition 4  
  sujets de requête 14  
dimensions  
  ambiguïté 48  
  de rôles 23  
  groupes sous forme de schémas en étoile 34  
  hiérarchies 31  
  identification 22  
  mesure 19, 33  
  modèle 8  
  ordinaires 8, 14, 19, 31  
  partage 19  
  sujets de requête 8  
dimensions conformes 35  
  faits multiples 39, 44  
dimensions de mesure 19  
  création 33  
  de rôles 23  
dimensions de rôles 23  
données de faits 30  
données dimensionnelles 29  
double comptage 22, 42, 48

## F

faits 33  
  ambiguïté 48  
  identification 22  
faits multiples, requêtes 8, 39, 44

## G

granularité 27  
granularité multiple, requêtes 8, 39, 44  
groupes sous forme de schémas en étoile 19  
  conformes multiples 35  
  création 34

## H

hiérarchies 4, 8, 31  
  multiples 31

## J

jointures  
  boucle 26

## L

liées, requêtes 22

## M

métadonnées DMR (relationnelles modélisées sous forme de  
  dimension) 31  
métadonnées importées  
  vérification 22  
modélisation dimensionnelle de métadonnées  
  relationnelles 31  
modélisation relationnelle, concepts 1

## O

objets ambigus 48  
objets de modèle  
  raccourcis 15  
opérations de calcul 17  
ordinaires, dimensions 8, 14, 19  
  création 31  
  de rôles 23  
  hiérarchies 31  
ordre des opérations de calcul 17

## P

partage, dimensions 19

## R

- raccourcis 15
- règles de cardinalité 2
- relations
  - 1-n 42
  - ambiguïté 23
  - cardinality 2
  - multiples valides 23, 26
  - niveaux de granularité 27
  - vérification 22
- relations de type 1-1 42
- relations de type 1-n 42
- relations récursives 27
- relations réflexives 27
- relations valides 23, 26
  - multiple 23
- requêtes 35
  - fait unique 37
  - faits multiples 8, 39, 44
  - granularité multiple 8
  - liées 22
  - sans fait 35
  - split 49
- requêtes à fait unique 37
- requêtes de faits croisés 22

- requêtes dimensionnelles 37
  - fait unique 37
  - granularité et faits multiples 39, 44
- requêtes fractionnées 49
- résolution
  - objets ambigus 48
  - requêtes fractionnées 49

## S

- schémas en étoile, concepts 29
- schémas en étoile conformes, groupes 35
- sources de données de type flocon de neige 29
- sources de données normalisées 29
- SQL 37
- sujets de requête
  - déterminants, 14
  - dimensions 8
  - groupes sous forme de schémas en étoile 34

## T

- tables Principale/Détails 30, 33
- type d'agrégation calculée 17