

Rapport radius

Description

L'objectif est de localiser les clients qui achètent un produit dans un rayon autour d'un emplacement spécifié comme dans l'exemple ci-dessous :

Your Zipcode	<input type="text" value="07088"/>	Product	<input type="text" value="BEER - AMSTEL - AMSTEL LIGHT BEER"/>	Miles	<input type="text" value="5"/>
Customer [zipcode_distances.tnl-Dive A]					
Customer	distance	Premise	County	City	
Totals					
CANDIDA BATISTA	3	On	NJ, Union	NJ, SPRINGFIELD	
J L D M INC	4	On	NJ, Essex	NJ, NEWARK	
ANTHONY'S PIZZA INC	4	On	NJ, Union	NJ, HILLSIDE	
STORAN INC.	4	On	NJ, Essex	NJ, NEWARK	
SONS OF ITAL REALIT	5	On	NJ, Essex	NJ, NEWARK	
W JOSEPH & R SUPERN	5	On	NJ, Essex	NJ, NEWARK	
BAYSTATE HOSPITALIT	5	On	NJ, Essex	NJ, NEWARK	
THE PITCHER'S MOUND	5	On	NJ, Essex	NJ, NEWARK	

Les prérequis sont :

- un Modèle indexé contenant les données de Produits et de Clients
- un Modèle indexé contenant les distances entre 2 codes postaux
- un script "Tunnel" Integrator
- un DivePlan pour manipuler les alias de colonnes afin d'avoir du texte convivial
- une Marque avec des QuickViews

Modèle indexé contenant les données de Produits(Products) et de Clients (Customer).

Il inclue les codes postaux des clients. Peut être créé à partir de votre Modèle Ventes existant :

- dimensions = { `Produit` },
- detail_dimensions= { `Client`, `Produit`, `Zipcode`, },
- string = { `Zipcode` }

Ce Modèle fonctionne comme une table SQL avec un index sur le Produit et fournit un accès rapide pour tous les clients qui achètent un produit.

Modèle indexé contenant les distances entre 2 codes postaux

Créez un fichier de codes postaux avec leur longitude et latitude. Vérifiez les données avec le fichier de carte zip.mid fourni par Dimensional Insight.

Zipcode	Lon	Lat
7001	-74.2702	40.5855
7002	-74.1068	40.6701
7003	-74.1871	40.8086
7004	-74.2976	40.8765
7005	-74.4215	40.9172
7006	-74.2854	40.8548
7008	-74.2335	40.5816
7009	-74.2287	40.8572

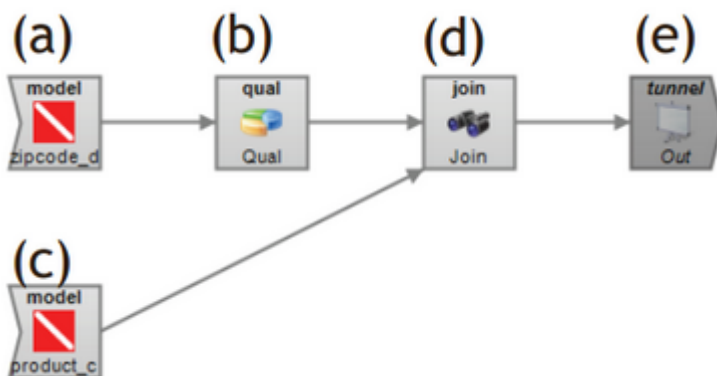
Effectuez une jointure du fichier sur lui-même et calculez les distances en utilisant la fonction miles(from_lat, from_lon, to_lat, to_lon).

Créez le Modèle indexé :

- dimensions = { `from_zipcode` },
- detail_dimensions = { `from_zipcode`, `to_zipcode`, `distance` },
- string = { `from_zipcode`, `to_zipcode` }

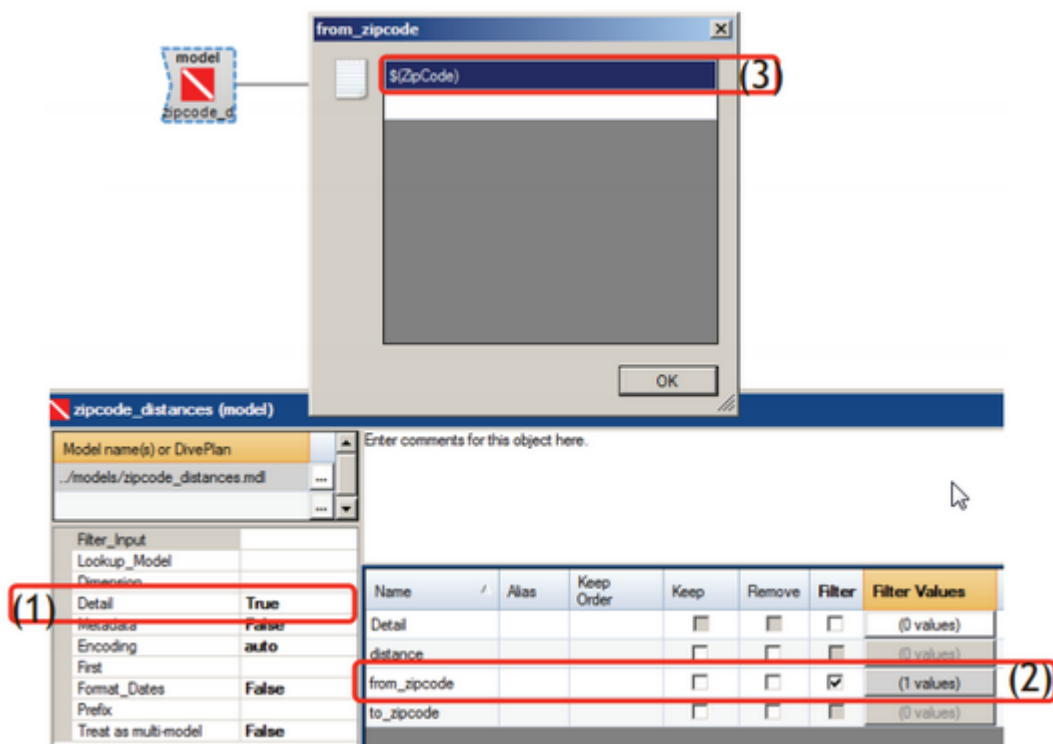
Ce Modèle fonctionne comme une table SQL avec un index sur From_ZipCode et fourni un accès rapide à toutes les valeurs ZipCode et les miles provenant de From_ZipCode. Ce Modèle doit être recréé si seulement de nouveaux Codes Postaux sont créés.

Il faut maintenant créer un script Integrator "Tunnel" dont voici le flux de données :



- (a) obtient les codes postaux et les distances
- (b) filtre les distances des codes postaux
- (c) obtient les produits achetés par les clients
- (d) jointure par Code Postaux
- (e) construction d'un Memory Model

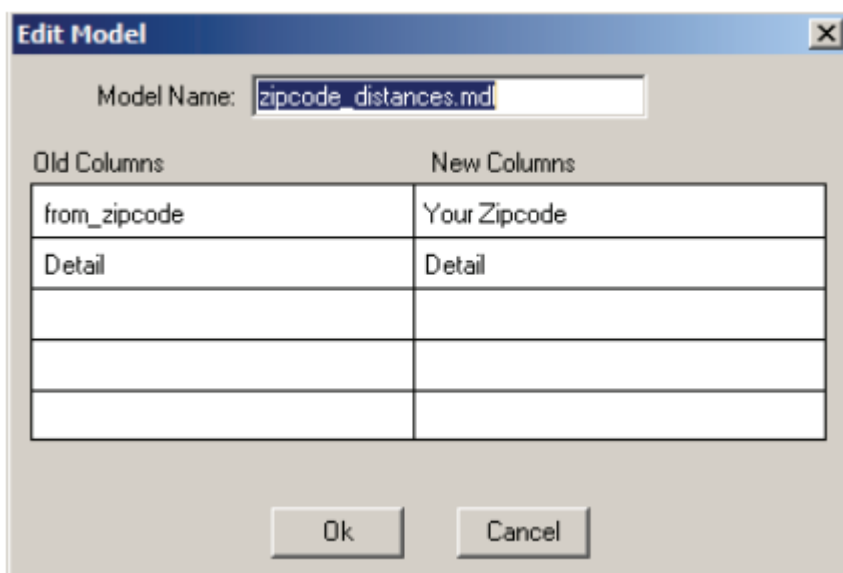
Lecture du Modèle indexé



- (1) Detail = True
- (2) Spécifier les colonnes de filtre
- (3) Spécifier les valeurs de filtre

DivePlan pour manipuler les alias de colonnes afin d'avoir du texte convivial

Utilisez (Pro)DiveMaster. Spécifiez les alias qui doivent apparaitre dans l'interface graphique.



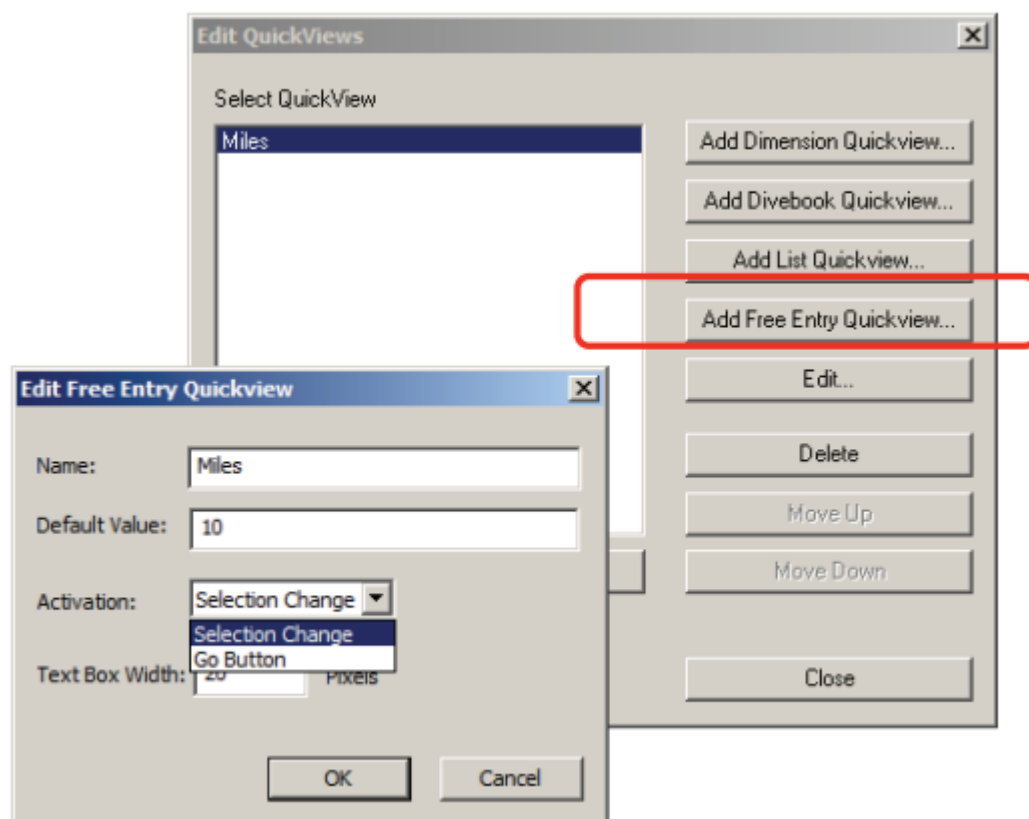
Marque avec des QuickViews

Plongez sur Customer, ajoutez des champs Infos.

Créez la QuickView "Votre Code Postal" (Your ZipCode) basée sur un Modèle externe et sur le DivePlan contenant des Alias.

Créez la QuickView "Produit" (Product) basée sur un Modèle externe : le Modèle Produit indexé.

Créez la QuickView "Miles". En version 6.4 la QuickView d'entrée libre pourra être utilisée. Dans les versions antérieures, il faudra d'abord créer un Modèle avec une Dimension Unique dont les valeurs iront par exemple de 1 à 1000 avec un incrément de 1, puis créer une QuickView de Modèle dont la source sera le Modèle créé à l'instant.



Remarque : vous pouvez remplacer la distance en miles par la distance en kilomètres en utilisant la fonction `kilometers(from_lat, from_lon, to_lat, to_lon)`.

Tags

1. DiveMaster
2. DivePlan
3. Diver
4. QuickView
5. script
6. Tunnel