

Oracle® Business Activity Monitoring

Enterprise Link Design Studio ユーザーズ・ガイド

10g (10.1.3.1.0)

部品番号 : B31883-01

2007 年 1 月

Oracle Business Activity Monitoring Enterprise Link Design Studio ユーザーズ・ガイド, 10g (10.1.3.1.0)

部品番号 : B31883-01

原本名 : Oracle Business Activity Monitoring Enterprise Link Design Studio User's Guide, 10g (10.1.3.1.0)

Copyright © 2002, 2006 Oracle. All rights reserved.

制限付権利の説明

このプログラム（ソフトウェアおよびドキュメントを含む）には、オラクル社およびその関連会社に所有権のある情報が含まれています。このプログラムの使用または開示は、オラクル社およびその関連会社との契約に記された制約条件に従うものとします。著作権、特許権およびその他の知的財産権と工業所有権に関する法律により保護されています。

独立して作成された他のソフトウェアとの互換性を得るために必要な場合、もしくは法律によって規定される場合を除き、このプログラムのリバース・エンジニアリング、逆アセンブル、逆コンパイル等は禁止されています。

このドキュメントの情報は、予告なしに変更される場合があります。オラクル社およびその関連会社は、このドキュメントに誤りが無いことの保証は致し兼ねます。これらのプログラムのライセンス契約で許諾されている場合を除き、プログラムを形式、手段（電子的または機械的）、目的に関係なく、複製または転用することはできません。

このプログラムが米国政府機関、もしくは米国政府機関に代わってこのプログラムをライセンスまたは使用する者に提供される場合は、次の注意が適用されます。

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

このプログラムは、核、航空産業、大量輸送、医療あるいはその他の危険が伴うアプリケーションへの用途を目的としておりません。このプログラムをかかるとして使用する際、上述のアプリケーションを安全に使用するために、適切な安全装置、バックアップ、冗長性（*redundancy*）、その他の対策を講じることは使用者の責任となります。万一かかるとしてプログラムの使用に起因して損害が発生いたしましても、オラクル社およびその関連会社は一切責任を負いかねます。

Oracle、JD Edwards、PeopleSoft、Siebel は米国 Oracle Corporation およびその子会社、関連会社の登録商標です。その他の名称は、他社の商標の可能性があり得ます。

このプログラムは、第三者の Web サイトへリンクし、第三者のコンテンツ、製品、サービスへアクセスすることがあります。オラクル社およびその関連会社は第三者の Web サイトで提供されるコンテンツについては、一切の責任を負いかねます。当該コンテンツの利用は、お客様の責任になります。第三者の製品またはサービスを購入する場合は、第三者と直接の取引となります。オラクル社およびその関連会社は、第三者の製品およびサービスの品質、契約の履行（製品またはサービスの提供、保証義務を含む）に関しては責任を負いかねます。また、第三者との取引により損失や損害が発生いたしましても、オラクル社およびその関連会社は一切の責任を負いかねます。

目次

	はじめに	v
	表記規則	vi
	このマニュアルの構成	vi
	Copyright Information from Group 1 Software	vii
	サポートおよびサービス	vii
第 1 章 :	概要	1
	データ・ウェアハウスおよびデータ・マート	2
	Oracle BAM Enterprise Link の概要	3
	Oracle BAM Enterprise Link の概要	3
	データ・フローのプランおよび Transform	5
	データ・マートへの移入	5
	集計ナビゲーション	6
	BaseView	6
	MetaView	6
	プログラミング機能	7
	プランの共有と分散	7
	高パフォーマンスのアーキテクチャ	8
第 2 章 :	Design Studio の起動	9
	Design Studio の起動	10
	接続設定の変更	11
	パスワードの変更	13
	Design Studio の環境	14
	Design Studio の構成要素	14
	Design Studio のアイテム	16
	プランおよび Snap の使用	17
	プランの破棄	19
	ユーザー・オプションの構成	19
	Design Studio の設定の変更	20
	Data Flow Service の切替え	21
	BaseView エディタの設定の変更	22

	Bin の設定の変更	23
	Design Studio のクローズ	24
第 3 章 :	BaseView の設計	25
	BaseView の概要	26
	BaseView での作業	26
	BaseView の作成	27
	Snap 用の BaseView の作成	34
	BaseView エディタの使用	35
	BaseView の変更	38
	BaseView のプロパティの変更	38
	表のプロパティの表示	39
	列のプロパティの表示	40
	物理名または表示名の表示	41
	BaseView への表の追加	42
	表の削除	43
	BaseView の結合	43
	別名表の作成	50
	表の自己結合	51
	BaseView の再同期化	52
第 4 章 :	MetaView の設計	59
	MetaView での作業	60
	MetaView の作成	60
	MetaView の保存	63
	MetaView の選択	63
	MetaView の削除	63
	MetaView のコピー	64
	MetaView の編集	64
	MetaView の名前の変更	65
	MetaView への BaseView の追加	65
	BaseView からの列の追加	66
	MetaView 内の Category の展開	67
	MetaView からの Part と Category の削除	67
	Part および Category の名前の変更	67
	Part および Category の再配置	68
	Category の追加	68
	Part のプロパティの表示	68
	制約の追加	71
	計算 Part の作成	73
第 5 章 :	データ・フロー・エディタの使用	77
	データ・フロー・エディタでのプランの表示	78
	データ・フロー・エディタでのズーム操作	79
	データ・フローの印刷	80
	複数のデータ・フローの作成	81
	データ・フロー・エディタからのプランの名前の変更	81
	Transform の使用	81
	SQL Query Transform	82

	Grid Transform.	85
	プランへの Transform の追加	85
	「Transform」メニューの使用	87
	ステップのプロパティの編集	90
	サブプランの使用	90
	プランでの反復の使用	95
	ステップの入力と出力の表示	101
	Transform の共有	102
第 6 章 :	プロパティでの作業	105
	概要	106
	Transform プロパティの使用	106
	「Step Settings」の表示	107
	USERINIT の設定	108
	カスタム・プラン・プロパティの使用	108
	カスタム・プラン・プロパティの表示	109
	カスタム・プラン・プロパティでの作業	109
	プラン・プロンプトの使用	117
	プラン・プロンプトの作成	118
	複数のパラメータの設定	120
	プラン・プロンプトの設定	122
	プラン・プロンプトの例の実行	125
第 7 章 :	StarMarts の設計.	131
	概要	132
	演習 : StarMart の構築	133
	StarView エディタの表示	133
	StarView の作成	134
	StarView の変更	144
	StarView の編集	144
	StarView の削除	156
	集計の環境のメンテナンス	157
	集計表のリフレッシュ	157
	移入のスケジューリング	158
	集計表の統計のリフレッシュ	159
	集計ポートフォリオの調整	159
	既存の集計表の使用	160
	集計移入プランのカスタマイズ	160
第 8 章 :	メタデータの統合	163
	メタデータの統合について	164
	OLE DB プロバイダとしての Enterprise Link の使用	164
	ソースとしての OLE DB プロバイダの使用	165
	Microsoft Repository での作業	167
	BaseViews のエクスポート	167
	Microsoft Repository オブジェクト階層	169
	Microsoft Repository からの変換のインポート	169
	DTS-Scripting Transform の使用	171

第 9 章 :	VBScript Transform	175
	Visual Basic Script Transform	176
	VBScript Transform	176
	VBScript Transform の使用	177
	ActiveX Automation のメソッドおよびイベント	178
	VBScript Transform イベント	179
	OLE イベント・プロシージャ	180
	レコード・オブジェクト API.....	181
	RecordDescriptor Object API.....	182
	ColumnDescriptor Object API	183
	ActiveX クライアント・サポート	185
	その他の API	186
	自動化サーバーの作成	186
	VBScript のフィールド・データ型	187
	VBScript のエラー処理	187
	VBScript のコンパイル・エラー	187
	VBScript のランタイム・エラー	188
	カスタム VBScript の例	189
	VBScriptInplace	189
	VBScriptCopy.....	190
	VBScriptSource	191
	VBScriptSink	192
	索引	193

はじめに

ここで説明する内容は次のとおりです。

- [表記規則](#)
- [このマニュアルの構成](#)
- [サポートおよびサービス](#)

表記規則

このマニュアルでは、次の表記規則を使用しています。

- キーボード上のキーは、[Esc]、[Enter] などの名前で示されます。
- ユーザーが指定する必要があるマシン名やその他の変数は、<machine name> などのように示されます。山カッコ (<>) は入力しないでください。
- ボタン名、リスト内の選択枝およびテキスト・フィールドの名前は、OK、ユーザー名などのように太字で示されます。
- パス、ファイル名およびコードは、C:¥Oracle BAM Enterprise Link などのように固定幅フォントで示されます。

このマニュアルの構成

このマニュアルでは、Design Studio の使用方法について説明します。Design Studio は、プランの設計を担当する Oracle BAM Enterprise Link 設計者向けのアプリケーションです。

このマニュアルで示す手順では、ユーザーは Microsoft Windows NT および Sun Solaris Operating System を熟知していると想定しています。

このマニュアルの各章の概要は、次のとおりです。

- 第1章「概要」では、データ・ウェアハウスとデータ・マート、および Oracle BAM Enterprise Link の概要を説明します。
- 第2章「Design Studio の起動」では、Design Studio アプリケーションの使用方法について説明します。
- 第3章「BaseView の設計」では、BaseView と、その使用方法について説明します。
- 第4章「MetaView の設計」では、MetaView と、その使用方法について説明します。
- 第5章「データ・フロー・エディタの使用」では、データ・フロー・エディタの使用方法について説明します。ここでは、Transform の使用方法についても説明します。
- 第6章「プロパティでの作業」では、Transform のプロパティの設定方法、カスタム・プラン・プロンプトの使用方法およびプラン・プロンプトの使用方法について説明します。
- 第7章「StarMarts の設計」では、StarMart と、その使用方法について説明します。
- 第8章「メタデータの統合」では、OLE DB プロバイダとしての Design Studio の使用方法と、Microsoft リポジトリの使用方法について説明します。
- 第9章「VBScript Transform」では、Visual Basic Script Transform の使用方法について説明します。

Copyright Information from Group 1 Software

Information in this document is subject to change without notice. Companies, names and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Group 1 Software, Inc.

Copyright 1996-2004 Group 1 Software, Inc. All rights reserved.

Trademark Notices:

Information Studio and Sagent are registered trademarks of Group 1 Software, Inc. The Sagent Logo, Sagent Design Studio, Flashcube, and StarMart are trademarks of Group 1 Software, Inc. All other trademarks are property of their respective companies.

Any provisions of the Sagent license related to ICU that differ from the IBM Public License 1.0 are offered by Sagent alone and not by any other party. The Source Code of the ICU program is available from Sagent upon written request. Further information regarding the ICU Program may be found at: <http://oss.software.ibm.com/icu/>.

You may make up to three copies of this publication, without the express written permission of Group 1 Software, for use at the authorized installation site. Reproduction and distribution outside of the licensed site is strictly prohibited. The contents of this Guide are subject to the confidentiality and non-disclosure provisions in your license agreement with Group 1 Software.

Group 1 products are warranted to perform as documented on hardware platforms running versions/releases of operating systems, subsystems, compilers, and facilities that are currently supported by their manufacturers. Please see your license agreement for additional details.

サポートおよびサービス

次の各項に、各サービスに接続するための URL を記載します。

Oracle サポート・サービス

オラクル製品サポートの購入方法、および Oracle サポート・サービスへの連絡方法の詳細は、次の URL を参照してください。

<http://www.oracle.co.jp/support/>

製品マニュアル

製品のマニュアルは、次の URL にあります。

<http://otn.oracle.co.jp/document/>

研修およびトレーニング

研修に関する情報とスケジュールは、次の URL で入手できます。

<http://www.oracle.co.jp/education/>

その他の情報

オラクル製品やサービスに関するその他の情報については、次の URL から参照してください。

<http://www.oracle.co.jp>

<http://otn.oracle.co.jp>

注意： ドキュメント内に記載されている URL や参照ドキュメントには、Oracle Corporation が提供する英語の情報も含まれています。日本語版の情報については、前述の URL を参照してください。

第 1 章

概要

ここで説明する内容は次のとおりです。

- データ・ウェアハウスおよびデータ・マート
- Oracle BAM Enterprise Link の概要

データ・ウェアハウスおよびデータ・マーケット

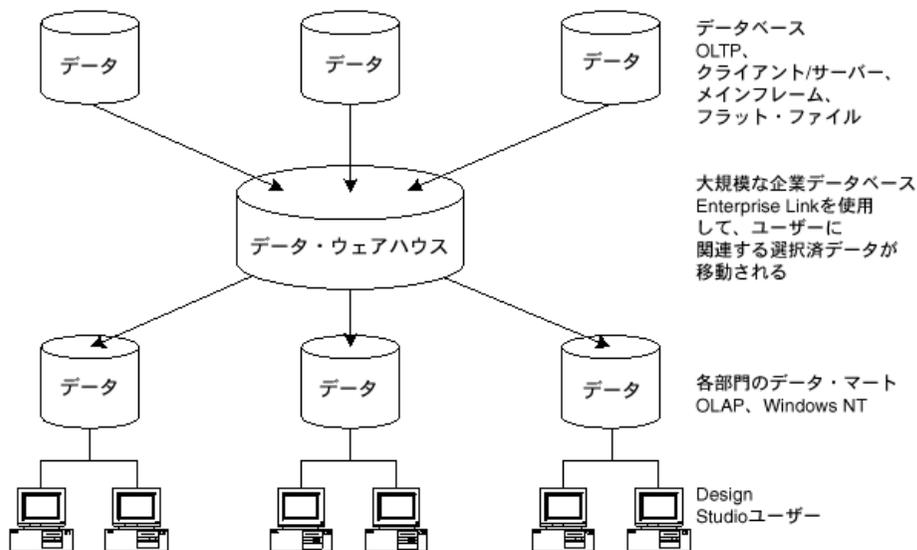
企業データは、オンライン・トランザクション処理（OLTP）という方法を使用して、業務系データベースに入力および収集されます。OLTP データベースは、レコードの挿入、更新および削除を対象にして最適化されますが、複雑な問合せ、一時的な問合せに備えた最適化は行われません。

データ・ウェアハウスは、OLTP データベースでのデータ分析およびその他の意思決定支援機能の実行に関連する問題に対応するために考案されました。企業は 1 つ以上の業務系データベースからデータを抽出し、特に問合せと分析向けに構築されている独立したデータ・ウェアハウスにそのデータを移動します。

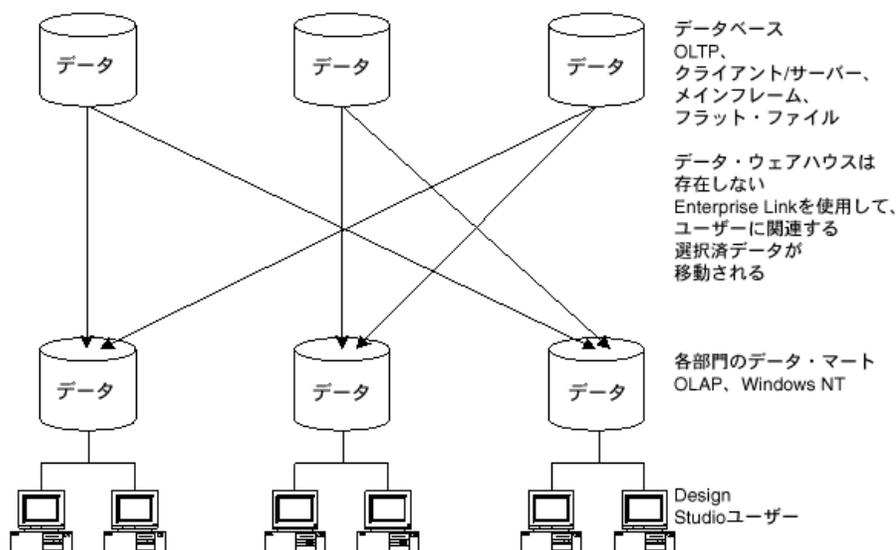
企業のデータ・ウェアハウスは増大し、複雑になるにつれ、問合せパフォーマンスが低下し、集中型モデルは、情報に対する簡単でタイムリなアクセスというユーザーの期待を満たせなくなります。これらのパフォーマンスの問題は、データを中央のデータ・ウェアハウスから、使用するユーザーと同じネットワークにあるサブジェクトベースの小規模なデータ・マーケットにステージングすることで対処できます。通常、データ・マーケットにはデータ・ウェアハウスのデータのサブセットが含まれていますが、データ・マーケットでは、多数の異種ソースからのデータを混在させ、中央のウェアハウスよりも対象範囲を拡大することもできます。

データ・マーケットは、アプリケーション固有の小規模なデータ・ウェアハウスで、戦略的なビジネス情報の分析に使用されます。データ・マーケットは、サブジェクト、部門、ワークグループ、地理または機能に応じて編成され、各部門に対してデータ・ウェアハウスの利点が提供されます。あるユーザー・グループのパフォーマンス要件が、他のグループのニーズに負の影響を与えることはありません。

次の図は、データが業務系データベースから集中型データ・ウェアハウスに移動された後、各部門のデータ・マーケットに移動される様子を示しています。



次の図は、集中型データ・ウェアハウス層を使用しなくても、企業がデータ・マートを使用できる様子を示しています。各部門のデータ・マートには、企業の業務系データベースから移入が直接行われます。



Oracle BAM Enterprise Link の概要

この項では、Oracle BAM Enterprise Link および Oracle BAM Design Studio の概要を説明します。この項では、Oracle BAM Enterprise Link のアーキテクチャと製品コンポーネントについて学習します。また、エンド・ユーザー・ビューの作成と意思決定支援データベースの構築に使用する Oracle BAM Design Studio の機能についても説明します。

Oracle BAM Enterprise Link の概要

Oracle BAM Enterprise Link は、統合された高性能な環境であり、複数のソースからの情報の抽出、意思決定支援のための情報の最適化、およびビジネス・ニーズに応じた形式での情報の配信を行うツールを提供します。Oracle BAM Enterprise Link では、クライアント・アプリケーションを介するユーザー・アクセス以外に、Web を介するユーザー・アクセスが提供されます。

Design Studio

Design Studio を使用すると、エンド・ユーザー向けのビジネス・ビューを定義し、データを抽出して変換し、それを1つ以上の意思決定支援データベースにロードできます。

Oracle BAM Design Studio では、ユーザーはメタデータを作成、インポートおよびエクスポートできます。Oracle BAM Design Studio で作成したビューは、データにアクセスして操作する必要があるエンド・ユーザーのビジネス層を定義します。また、Oracle BAM Design Studio ユーザーは、視覚的なデータ・フローを使用した追加の処理を行って、データに対する複雑な問合せを作成することもできます。これには、組込み式のコンポーネント以外に、これらのコンポーネントのカスタマイズ・バージョン、VBScript、PerlScript または JavaScript コンポーネント、および Transform Software Developer's Kit で作成される C++ コンポーネントが含まれています。

Enterprise Link Server に接続すると、Oracle BAM Design Studio ユーザーは、データ・マートまたは意思決定支援データベースを作成するための移入プランを作成できます。Oracle BAM Design Studio には、複数のソースから 1 つ以上のターゲットに対して抽出、変換およびロードを行うための特別なコンポーネントが含まれています。

Enterprise Link Server

Enterprise Link Server には、リポジトリおよび Data Flow Service が含まれています。リポジトリは、メタデータと、結果を取得、操作および表示するための指示などの特定のアイテムが格納されるデータベースです。Data Flow Service は、Oracle BAM Enterprise Link 環境でクライアント / サーバー間のすべての対話を仲介するサーバー・アプリケーションです。Data Flow Service は、インストールする Enterprise Link Server に応じて、データへのアクセスまたはデータのロード（あるいはその両方）を目的に構成できます。

Data Flow Service およびリポジトリは、複数層アーキテクチャで中間層を構成します。追加の Data Flow Service の使用を許諾して、追加ユーザー用に環境を拡張したり、複雑な操作のパフォーマンスを向上させたり、個別サーバーに対するロードおよびアクセス作業を実行します。また、リポジトリは、情報ニーズが異なるユーザー向けに、または開発、テスト、運用など、異なる環境向けに設計されたものを複数所有することもできます。

WebLink

WebLink では、Web ブラウザから Oracle BAM Enterprise Link 環境内のアイテムへの高速アクセスが提供されます。ユーザーは、イントラネットまたはインターネットを介して結果を取得し、それを標準の WebLink ページまたはカスタマイズしたページで表示できます。

Oracle BAM Enterprise Link Admin

Oracle BAM Enterprise Link Admin は、Enterprise Link 環境全体を管理するための管理アプリケーションです。Oracle BAM Enterprise Link Admin では、単一のインタフェースから複数の Enterprise Link Server を管理できます。実行可能なタスクには、リポジトリと Data Flow Service の構成、ユーザーとグループのセキュリティの定義、オブジェクトとメタデータの管理などがあります。

データ・フローのプランおよび Transform

Oracle BAM Enterprise Link におけるすべての処理は、一般的で視覚的なメタファであるデータ・フローを使用して定義されます。データ・フローは、プランと呼ばれる複数ステップのプロセスを介するデータのフローを制御するためのグラフィカルなツールです。プランは Oracle BAM Design Studio から実行されます。

プランは、単一のテンプレートまたはフィルタではなく、多数のステップで構成されます。1つの変換ステップ（他の製品ではその実行に複数のパスが必要）で、多数の操作を実行できます。データ・フローのアーキテクチャを使用することで、プランにおける複数のソースおよびターゲットの存在が可能になります。ソースが複数あっても、ロード前に中間データベースでマージする必要はありません。

データ・マートを構築するデータ設計者および DBA から、データ・マートの情報にアクセスして分析するエンド・ユーザーに至るまで、すべての Oracle BAM Enterprise Link ユーザーが同じデータ・フロー・インタフェースを使用してタスクを実行します。プランはモジュール式に設計されるため、プロセスの視覚化と変更が可能です。つまり、ユーザーは短時間で製品を学習し、実行するタスクに集中できるようになります。

データ・フローのステップを **Transform** と呼びます。Transform は、次のタイプの操作を実行するための組込み式のプログラム・モジュールです。

- プランにおけるデータのフロー制御
- データの分析
- データ・マートへの移入とデータの変換
- レコード・レベルでのデータ操作のカスタマイズ
- データの表示

データ・マートへの移入

企業でデータ・ウェアハウスをすでに実装しているか、または多様なソースから情報を収集しているかに関係なく、Oracle BAM Design Studio では、重要な情報をデータ・マートまたはデータ・ウェアハウスに迅速かつ正確に移動するために必要となるツールが提供されます。

データ・フロー・プランを作成して、意思決定支援データベースにデータを移動します。Oracle BAM Design Studio および Enterprise Link Server では、OLTP スキーマからスター・スキーマにデータを再構成するための移入 Transform が提供されます。データをロードする前に Grid に表示して、それが正しいことを確認できます。移入プランを定期的に行うようにスケジューリング設定すると、データを自動的に更新できます。

StarView エディタは重要な移入機能です。これを使用して、データ・マートで集計表を作成および管理し、問合せパフォーマンスをさらに向上させます。また、ユーザー独自の VBScript または C++ Transform コンポーネントを追加して、プランを再構成する移入およびデータをカスタマイズすることもできます。

集計ナビゲーション

Oracle BAM Design Studio では、問合せパフォーマンスを向上させるために集計を作成および管理する高度な環境が提供されます。StarView エディタおよび「StarView」ウィザードを使用すると、Oracle BAM Design Studio から集計を作成および管理できます。

統合されたソリューションの一部として、StarView では、メタデータの作成における高い柔軟性と拡張性が提供されます。集計用にデータ・マートを最適化すると、ナビゲーションに備えてメタデータが Enterprise Link クライアント・アプリケーションに渡されます。「StarView」ウィザードでは、ユーザーに対して透過的な集計ナビゲーションが提供されます。Oracle BAM Design Studio のプランは、集計を認識し、データへの最短パスを自動的に採用します。

BaseView

Design Studio では、BaseView を介して物理的なデータベース・スキーマを表示およびアクセスします。BaseView は、表、列、結合など、データベース・オブジェクトのためのメタデータ層です。Oracle BAM Design Studio で使用する高レベルのメタデータは、BaseView から作成します。BaseView では、基礎となる物理的なデータベースでの結合に影響を与えることなく、表間の追加の結合を定義できます。

BaseView エディタを使用して、BaseView の作成と編集、および基礎となるデータベースのスキーマの表示を行います。同じデータベースから異なる BaseView を作成できます。データベース管理者用のビューにはシステム表を含め、ユーザー用のビューにはデータ表のサブセットを含めるような場合があります。

MetaView

MetaView は、論理的なメタデータ層であり、1つ以上の BaseView の最上部に作成されます。MetaView は、エンド・ユーザーに表示されるソース・データのビューを定義します。MetaView では、データベース・アーキテクチャの複雑さをユーザーが意識することはなく、ビジネス・ニーズおよびボキャブラリに基づくビューが提供されます。同じデータの複数の MetaView を作成して、異なるユーザー・グループのニーズを満たすことができます。

MetaView は、BaseView によって表現される基礎となる構造に基づいています。同じ MetaView の複数の BaseView からの表および列は結合できます。MetaView を編集して、アイテムを並び替えおよび削除できます。また、BaseView から表および列をドラッグして、MetaView にそれらを追加できます。さらに、SQL 式を使用して MetaView に計算フィールドを作成し、計算を実行したり、選択制約を確立できます。

プログラミング機能

Oracle BAM Enterprise Link には、組織に特有のデータ変換処理を目的とした、強力なツール群が含まれています。

■ Transform Software Developer's Kit (SDK)

Transform SDK は、Transform を記述するための Application Program Interface (API) です。Transform オブジェクトは、プランでユーザー入力を解析する DLL です。プランの機能を拡張する必要があるアプリケーション開発者は、カスタム Transform オブジェクトを作成できます。Transform アーキテクチャは、再使用可能なコンポーネントを定義するためのオープン・スタンダードである Microsoft Component Object Model (COM) に準拠しています。

Transform SDK は、インタプリタ型スクリプト言語またはプログラムの動作のみを制御するカスタム・アプリケーション・ビルダーより強力です。C++ API を使用して作成する Transform は、Oracle BAM Enterprise Link 環境に完全に統合されます。データ・マイニング・ツールまたはクレンジング・ツールは、Transform SDK を使用して作成し、Oracle BAM Enterprise Link 環境に統合できる COM オブジェクトの代表的な例です。

Transform SDK を使用して Transform を作成する方法については、『Transform Programmer's Guide』を参照してください。

■ VBScript

VBScript エンジン（サーバー・システム用に最適化されている Microsoft Visual Basic のサブセット）は、Oracle BAM Enterprise Link に組み込まれています。VBScript Transform は、個々の小規模なタスクをカスタマイズするために作成します。

VBScript Transform を使用し、レコード・レベルでデータを操作します。SQL とは異なり、VBScript Transform は Enterprise Link Server で一度に 1 レコードを処理します。このため、ユーザーは結果セットの各レコードの状態を変更できます。

プランの共有と分散

リポジトリには、そのリポジトリのユーザーが作成したすべてのプランが含まれます。各プランは、それを作成したユーザーが所有します。プランは、パブリッシュまたは自動サブスクリプションすることで、使用しているリポジトリの他のユーザーと共有できます。ユーザーがプランをサブスクリプションすると、そのユーザーは情報の変更を自動的に受信するため、ワークグループ全体で一貫性が保持されます。

また、プランはファイルにエクスポートしたり、別のリポジトリにインポートできるため、リポジトリが異なるユーザーとプランを共有することもできます。さらに、Oracle BAM Design Studio で電子メールを統合して使用することで、プラン・ファイルを他のユーザーに送信できます。

高パフォーマンスのアーキテクチャ

データ・マートのアーキテクチャは、集中型データ・ウェアハウスの利点と、分散型のサブジェクトベースのデータ・マートのパフォーマンスを兼ね備えています。

■ マルチスレッド

Oracle BAM Enterprise Link アーキテクチャは、32-bit Windows Operating Systems の利点を最大限に活用しています。Oracle BAM Enterprise Link におけるマルチスレッディングおよびマルチタスキングのサポートによって、サーバー・タスクを並行して実行できます。

■ SQL の最適化と拡張性

ほとんどのプランは、SQL 問合せで開始します。Transform をプランに追加すると、結果に対して追加の操作を実行できます。Data Flow Service はプランと、そこに含まれている指示を実行します。データベース・サーバーが操作を実行した方がより効果的である場合、Data Flow Service はその操作を SQL レベルにプッシュ・バックします。

複数のソースおよびターゲットを単一のプランで使用できるため、1つのプランの同じ SQL 実行で複数の意思決定支援データベースをロードできます。

■ スケジューリング

プランは、ユーザーが介入しなくても特定の時間に実行するようにスケジュール設定できます。データ・マートへの移入を行うプランや、複雑な SQL 操作を実行するプランは、サーバー・パフォーマンスに影響を与えないように、使用率が低い時間帯に実行するようにスケジュール設定できます。スケジュール済のアイテムの実行が完了すると、Data Flow Service によって、結果が使用可能であることを通知する電子メールが送信されるか、または通知がログ・ファイルに記録されます。

■ キャッシング

Data Flow Service は、サーバー・マシンでキャッシュのプールを保持します。ユーザーが取得するすべての結果セットは、キャッシュに格納されます。別のユーザーが同じアイテムをリクエストした場合、Data Flow Service ではメモリーからそのアイテムを直接取得できます。Oracle BAM Enterprise Link 管理者は、キャッシュ済アイテムにアクセスするユーザーを制御できます。

■ Search Engine

Search Engine を使用すると、リポジトリ内のアイテムの検索が可能になります。検索は、キーワード、パブリッシュ属性およびサブスクリプション属性に基づいて実行できます。

Design Studio の起動

ここで説明する内容は次のとおりです。

- [Design Studio の起動](#)
- [Design Studio の環境](#)
- [ユーザー・オプションの構成](#)
- [Design Studio のクローズ](#)

Design Studio の起動

Design Studio はクライアント・アプリケーションであるため、ログインするには、リポジトリ・データベース・サーバーと Data Flow Service を実行している必要があります。

Design Studio を起動するには、次の手順を実行します。

1. 「スタート」メニューから、「プログラム」→「Iteration」→「Design Studio」を選択します。

「Design Studio」ログイン・ダイアログ・ボックスが表示されます。



2. Oracle BAM Enterprise Link ユーザー名とパスワードを入力して、Design Studio にログインします。
3. このセッションの接続パラメータを表示する場合は、「Advanced」をクリックします。

詳細なログイン・ダイアログ・ボックスが表示されます。これらの値を変更する方法については、[11 ページの「接続設定の変更」](#)を参照してください。

4. 「OK」をクリックします。

Design Studio のメイン画面が表示されます。

リポジトリでユーザーとして構成されていない場合は、エラー・メッセージが表示されます。このメッセージについて Oracle BAM Enterprise Link 管理者に連絡し、リポジトリに追加してもらいます。

新しいリポジトリに接続されている Design Studio を最初に起動するユーザーは、初期 BaseView と Snap 用の BaseView を作成する必要があります。

Design Studio 内のデータにアクセスできるようになるのは、1つの BaseView と1つ以上の MetaView を作成し、これらのオブジェクトに適切な権限を設定した後でのみです。詳細は、[27 ページの「BaseView の作成」](#)および [34 ページの「Snap 用の BaseView の作成」](#)を参照してください。権限の設定については、『Oracle Business Activity Monitoring Enterprise Link 管理者ガイド』を参照してください。

接続設定の変更

ログインする際に、Design Studio のそのときのセッションの接続設定を変更できます。たとえば、レジストリに格納されているデフォルト値とは異なるリポジトリまたは Data Flow Service を指定できます。定義した設定は、そのセッション以降は破棄するか、またはデフォルトのログイン設定として保存できません。

注意： また、Design Studio で「**Tools**」→「**Options**」を選択して、Data Flow Service 設定にアクセスすることもできます。

接続設定を変更するには、次の手順を実行します。

1. Design Studio のログイン・ダイアログ・ボックスで、「**Advanced**」ボタンをクリックします。

詳細なログイン・ダイアログ・ボックスが表示されます。

The screenshot shows the 'Design Studio' login dialog box with the 'Advanced' tab selected. The dialog is divided into two main sections: 'Data Flow Service' and 'Repository'.

- Data Flow Service:**
 - User Name: jsmith
 - Password: [masked]
 - Machine: Rocko
 - Port Number: 31857
 - Protocol: Sockets, Named Pipes
 - Delay connection until needed
- Repository:**
 - Server Type: SQL Server (dropdown)
 - Server Name: Behemoth
 - Database: samples_repository
 - User ID: jsmith
 - Password: [masked]

At the bottom, there is a checkbox for 'Save these Data Flow Service and Repository settings as my default login settings.' and buttons for 'Advanced <<', 'OK', and 'Cancel'.

2. 異なるネットワーク・マシンにインストールされている Data Flow Service を使用するには、「Machine」フィールドに新しいマシン名を入力します。
3. このクライアント・マシンが Data Flow Service と通信するためのネットワーク・プロトコルを変更するには、「Sockets」または「Named Pipes」のいずれかをクリックします。

これは、Data Flow Service がクライアント・リクエストのリスニングに使用するプロトコルと同じである必要があります。詳細は、『Oracle Business Activity Monitoring Enterprise Link 管理者ガイド』を参照してください。

4. Data Flow Service と通信するためのネットワーク・ポートを変更するには、「Port Number」フィールドに現在使用されていない値を入力します。

ソケットの場合、これは、Data Flow Service がクライアント・リクエストのリスニングに使用するポート番号と同じである必要があります。詳細は、『Oracle Business Activity Monitoring Enterprise Link 管理者ガイド』を参照してください。

名前付きパイプの場合、この数値は、システム生成されるパイプ名に付加され、名前の一意性が保証されます。

5. Data Flow Service に接続することなく Design Studio にログインするには、「Delay connection until needed」をクリックします。プランを実行またはスケジュール設定するには、接続が必要になります。プランを実行またはスケジュール設定する際に、Design Studio によってネットワーク接続が自動的に確立され、Design Studio を終了するまで保持されます。

6. デフォルトとは異なるリポジトリを使用するには、ダイアログ・ボックスの右側でデータベース接続パラメータを指定します。

- Server Type: Microsoft SQL Server、Oracle、Sybase などのデータベース・タイプ
- Server Name: リポジトリが格納されているサーバーの名前か、または Oracle のホスト文字列 (domain.instanceName)
- Database: リポジトリ・データベースの名前 (Oracle リポジトリの場合は不要)
- User ID: リポジトリ・データベースで select、insert、update および delete 操作を実行する権限のあるログイン
- Password: リポジトリ・データベース・ログインのパスワード

7. このダイアログ・ボックスの値をデフォルトのログイン設定として使用する場合は、設定を保存するためのチェック・ボックスを選択します。

8. 「OK」をクリックします。

ダイアログ・ボックスの設定を使用して Design Studio へのログインが行われ、メイン画面が表示されます。

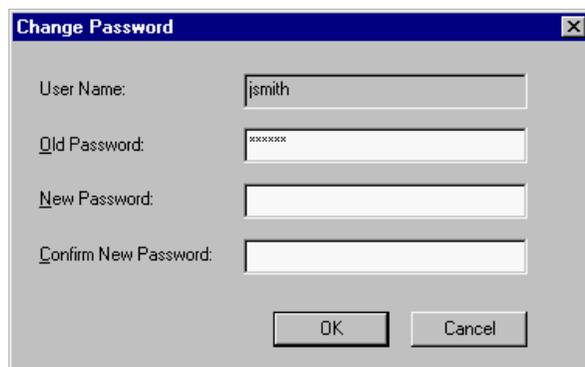
リポジトリでユーザーとして構成されていない場合は、エラー・メッセージが表示されます。このメッセージについてシステム管理者に連絡し、リポジトリに追加してもらいます。

パスワードの変更

Oracle BAM Enterprise Link 管理者は、ユーザーのパスワードに影響する属性を設定できます。これらの属性には、アカウントの無効化、ユーザーに対するパスワード変更の要求、および空白のパスワードの許可が含まれます。また、Oracle BAM Enterprise Link 管理者は、パスワードをいつでも変更できるアカウント権限を付与したり、パスワードの有効期限を設定することもできます。パスワードを変更するには、次の手順を実行します。

1. 「File」 → 「Change Password」 を選択します。

「Change Password」 ダイアログ・ボックスが表示されます。

A screenshot of the 'Change Password' dialog box. The dialog has a title bar with 'Change Password' and a close button. It contains four text input fields: 'User Name:' with 'jsmith' entered, 'Old Password:' with 'xxxxxxx' entered, 'New Password:' which is empty, and 'Confirm New Password:' which is empty. At the bottom, there are 'OK' and 'Cancel' buttons.

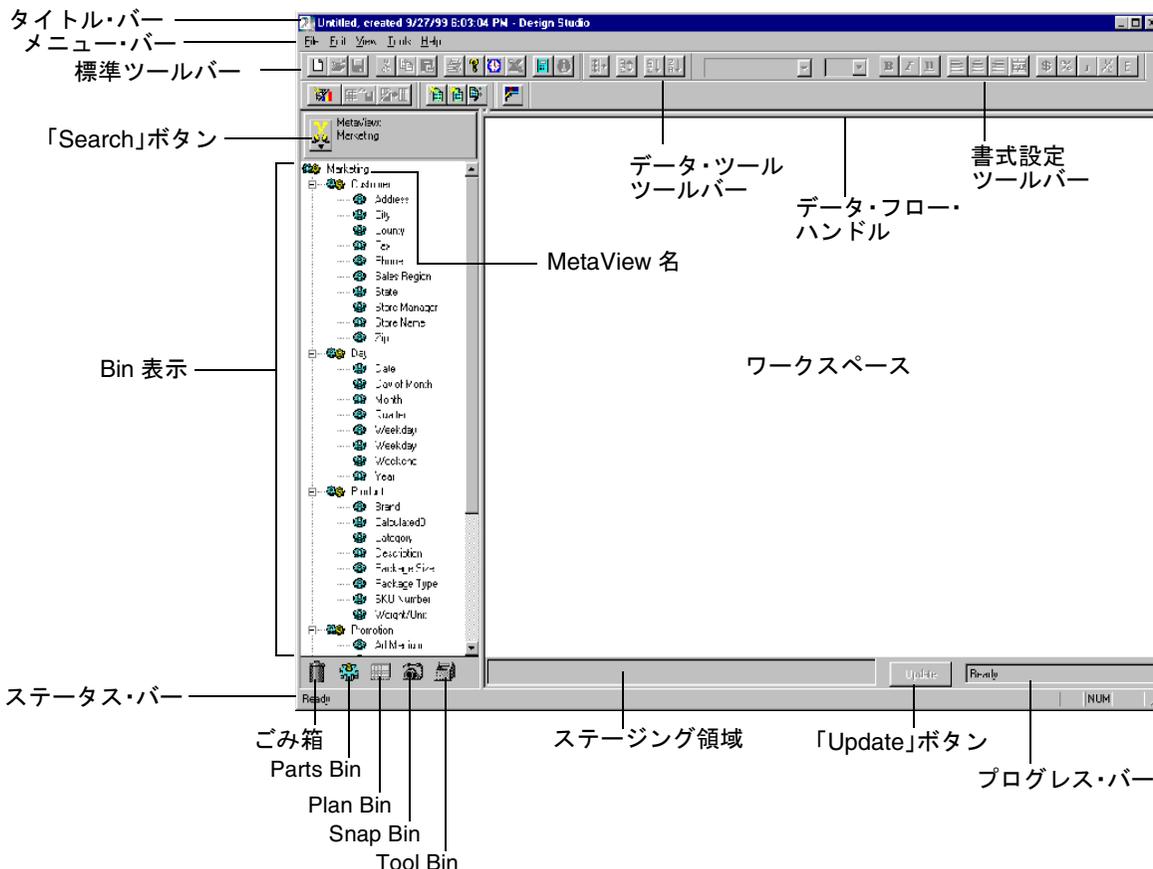
2. 「Old Password」 フィールドに現行のパスワードを入力します。
3. 「New Password」 フィールドに新しいパスワードを入力します。
4. 「Confirm New Password」 フィールドに新しいパスワードを再入力します。
5. 「OK」 をクリックします。

Design Studio の環境

この項では、Design Studio の構成要素について説明します。

Design Studio の構成要素

Design Studio を起動すると、メイン画面が表示されます。



タイトル・バー

タイトル・バーには、アプリケーションの名前と、現在オープンしているアイテムの名前が表示されます。

メニュー・バー

メニュー・バーには、アクティブな情報に対して使用できるコマンドがグループ化されています。

標準ツールバー

標準ツールバーには、一般的なメニュー・コマンド用のショートカット、スケジュール用のボタン、および結果を Excel に送信するためのボタンがあります。ボタンにポインタを置くと、ツールのヒントとボタンの名前が表示されます。

データ・ツール・ツールバー

データ・ツール・ツールバーのボタンによって、フィルタおよびソートのダイアログ・ボックスが表示されます。

書式設定ツールバー

書式設定ツールバーには、プラン結果のテキストと数値を書式設定するためのボタンがあります。

データ・フロー・ハンドル

データ・フロー・ハンドルを使用してワークスペース・パネルをプルダウンすると、データ・フロー・エディタが表示されます。ここに、現行のプランがグラフィカルに表現されます。

「Search」ボタン

検索ボタンを使用して、Parts Bin 内の MetaView を選択したり、プラン、Snap などの他の Bin 内のアイテムを検索します。

Bin 表示

Bin 表示は、各 Bin の内容が表示される領域です。

ワークスペース

ワークスペースは、プランおよび Snap を作成、表示および実行するための領域です。

ステータス・バー

ステータス・バーには、ポインタを置いたアイテムについての情報が 1 行で表示されます。

ステージング領域

ステージング領域は、開発中、実行中または実行後のプランを保持する一時的な格納領域です。この領域を使用すると、同時に複数のプランおよび Snap で作業できます。

「Update」および「Stop」ボタン

このボタンでプランの実行を開始したり、実行中のプランを停止します。

タブ

ワークスペースのタブは、プランがワークスペースにあるときは常に表示されます。データ・フロー・シンクごとに、1 つのタブが表示されます。現行のプランにデータ・フロー・シンクがない場合は、空白のタブが表示されます。

プログレス・バー

プログレス・バーには、ワークスペースに表示されているプランのステータスが示されます。

Design Studio のアイテム

Design Studio で使用する主なアイテムは、Part、プラン、Snap および Transform です。これらの各アイテムに固有の Bin ボタンがあり、これらのボタンはステータス・バーの上に表示されます。Bin を開くと、その内容が Bin ボタンの上の領域に表示されます。

Category



Account

Category は、Part を論理的な関係にグループ化します。Category は、データベース内の表でも、アイテムの新しいグループでもかまいません。

Part



Amount

Part は、プランの基本構成ブロックです。Part は、データベース内のフィールドでも、計算フィールドでもかまいません。Part は、Category にグループ化されます。

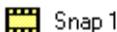
プラン



Plan 1

プランは、データを取得するために使用する一連の指示です。簡単な例としては、データベースから複数のフィールドを取得して、それらを列表示することが挙げられます。より複雑な例には、それらのフィールドでの計算、フィルタまたはソートがあります。

Snap



Snap 1

Snap は、プランからの結果を保存する方法です。これによって、異なる時点からの情報をアーカイブできます。

Transform



Join

Transform は、プランにステップを追加します。これらは、Design Studio ユーザーおよび Information Studio のパワー・ユーザーの Tool Bin で使用できます。Transform は、SQL スクリプト、組込みファンクションまたはカスタム・プログラムにすることができます。Transform は、SQL を超えたプランの処理能力の拡張を実現します。

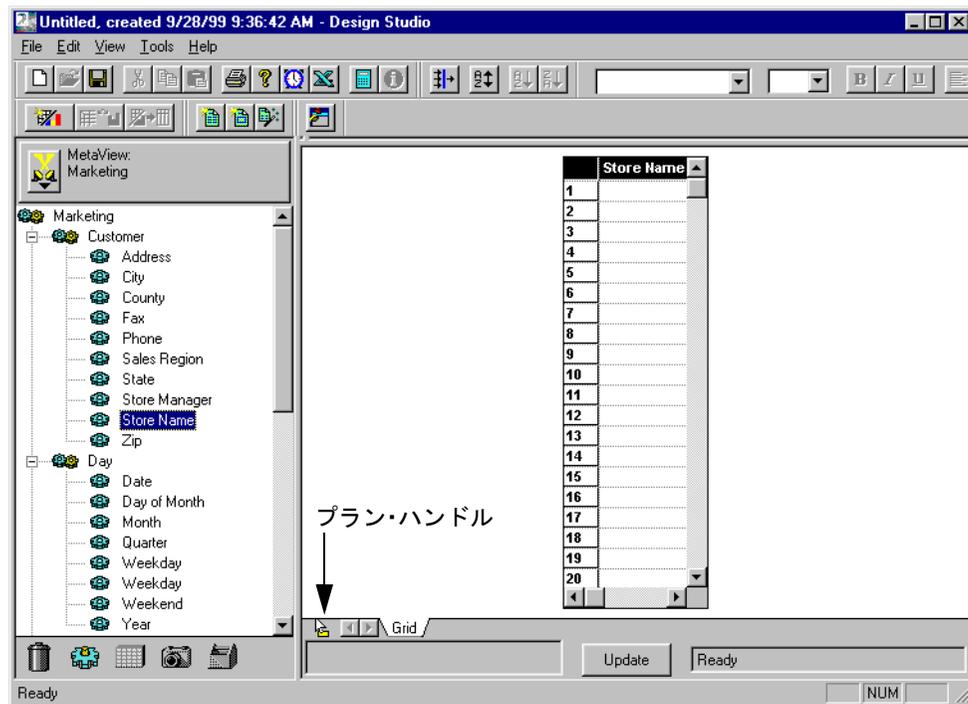
プラン・ハンドル



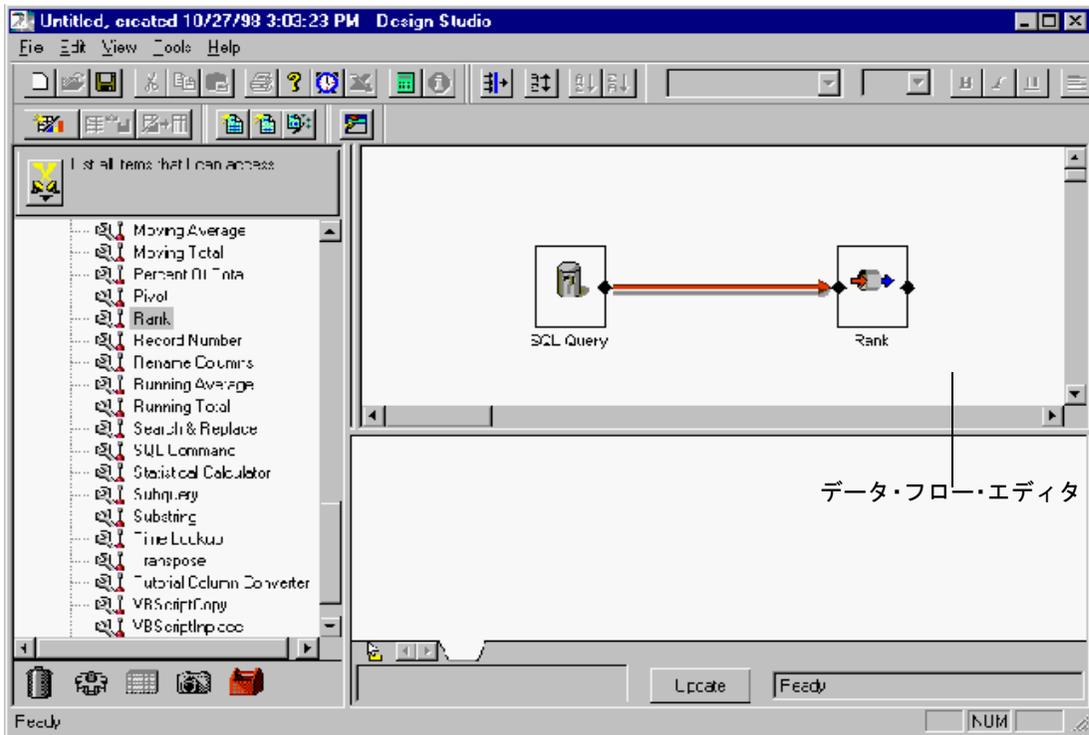
プラン・ハンドルは、Bin またはステージング領域にプランをドラッグするために使用されます。

プランおよび Snap の使用

プランは、データに対する取得操作および実行操作のための一連の指示です。指示は、Transform（プランのモジュール式の構成ブロック）群に含まれます。プランの作成は、Part をワークスペースにドラッグするか、または Transform をデータ・フロー・エディタにドラッグして行います。



Tool Bin には、Design Studio とともに提供される Transform と、自分や他のユーザーが作成したカスタム Transform が含まれます。プランは、Tool Bin からデータ・フロー・エディタに Transform をドラッグして作成できます。



プランを実行すると、そこに含まれている指示が実行されます。プランがデータを表示するシンクで終了した場合は、その結果が表示されます。その指示をプランとして保存したり、その結果を Snap として保存できます。

プランおよび Snap は、リポジトリに格納されます。これらには、Design Studio の Plan Bin および Snap Bin からアクセスします。

プランまたは Snap を保存するには、次の手順を実行します。

- プラン・ハンドルを Plan Bin または Snap Bin にドラッグします。

Snap は、プランが取得したデータを含むデータベース表です。プランの結果を Snap として保存すると、その表は、Snap BaseView および Snap MetaView が参照するデータベースに保存されます。他のデータベース表の場合と同様に、Snap 表に対して問合せを実行できます。Snap BaseView の詳細は、[34 ページの「Snap 用の BaseView の作成」](#)を参照してください。

プランの破棄

ワークスペースから Grid を削除するには、それをごみ箱にドラッグするか、または消去します。

ワークスペースの消去および現行プランの破棄を実行するには、次のいずれかを実行します。

- プラン・ハンドルをごみ箱ボタンにドラッグします。
- 「File」 → 「Clear Workspace」を選択します。この場合、プランはごみ箱に格納されずに破棄されます。

また、プランを破棄せずにワークスペースを消去することもできます。

- 標準ツールバーで「New Plan」ボタンをクリックするか、または「File」 → 「New Plan」を選択します。

プランがステージング領域に移動します。ステージング領域は、ワークスペースの底部にあります。ステージング領域の使用の詳細は、Oracle BAM Design Studio のオンライン・ヘルプを参照してください。

ユーザー・オプションの構成

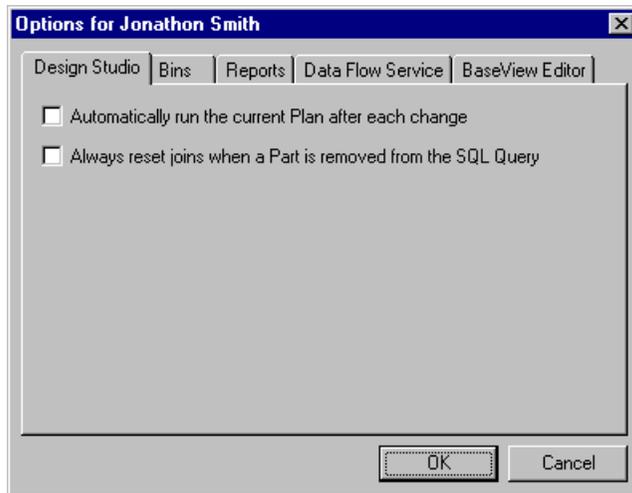
ログイン後に、「Options」ダイアログ・ボックスの設定を変更して、Design Studio の動作をカスタマイズできます。ダイアログ・ボックス内の設定は、Design Studio セッションに影響を与え、変更するまで保存されます。「Options」ダイアログ・ボックスでは、次の変更を実行できます。

- プランが変更された際にそのプランを自動的に実行する
- Part が削除された際に結合を常にリセットする
- 削除プロンプトをカスタマイズする
- 異なる Data Flow Service マシンに接続する

「Options」ダイアログ・ボックスを表示するには、次の手順を実行します。

➤ 「Tools」 → 「Options」 を選択します。

「Options」ダイアログ・ボックスを表示します。



Design Studio の設定の変更

「Design Studio」タブで、変更後にプランを自動的に実行するように指定できます。また、問合せから Part が削除される際にそのプランに使用されている結合パスをリセットすることもできます。

変更した際にプランをすぐに実行するには、次の手順を実行します。

➤ 「**Automatically run the current Plan after each change**」チェック・ボックスを選択します。

Part の追加、フィルタ条件の変更、Transform の追加、データ・フロー内のコネクタの削除など、どのような変更でもプランは実行されます。プランの変更はすぐに検出されます。プランは、Plan Bin に保存する必要はありません。

Part が削除されるたびに結合をリセットするには、次の手順を実行します。

➤ 「**Always reset joins when a Part is removed from the SQL Query**」チェック・ボックスを選択します。

これを設定していない場合、結合は無効になったときにのみリセットされます。Part の削除とは、データ・フロー内の SQL Query ステップから Part を削除することです。これには、表示シンクからの Part の削除または非表示は含まれません。

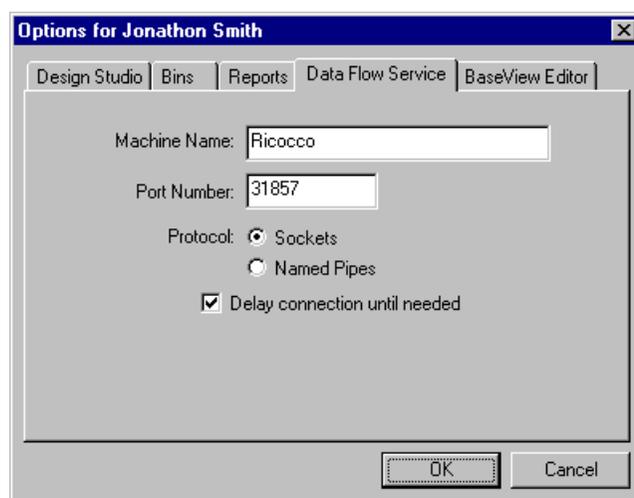
Data Flow Service の切替え

複数の Enterprise Link Server のライセンスがあり、それらをネットワークにインストールしている場合は、Design Studio の実行中に別の Data Flow Service に切り替えることができます。Data Flow Service のタイプによって、Tool Bin の内容が決定されます。Data Flow Service は、ロードまたはアクセス（あるいはその両方）に対して有効にできます。Data Flow Service を切り替える場合は、ワークスペースまたはステージング領域にプランが存在していないことを確認します。

注意： Data Flow Service の設定は、Design Studio のログイン・ダイアログ・ボックスで「**Advanced**」ボタンをクリックして変更することもできます。詳細は、[11 ページの「接続設定の変更」](#)を参照してください。

異なる Data Flow Service に接続するには、次の手順を実行します。

1. 「Data Flow Service」タブをクリックします。



2. 異なるネットワーク・マシンにインストールされている Data Flow Service を使用するには、「Machine Name」フィールドに新しいマシン名を入力します。
3. このクライアント・マシンが Data Flow Service と通信するためのネットワーク・プロトコルを変更するには、「Sockets」または「Named Pipes」のいずれかをクリックします。
4. Data Flow Service と通信するためのネットワーク・ポートを変更するには、「Port Number」フィールドに現在使用されていない値を入力します。

5. タスクの実行に Data Flow Service への接続が必要になるまで、その接続を遅延させるには、「**Delay connection until needed**」を選択します。

このオプションによって、現行の Data Flow Service 接続は終了されますが、新しい接続はプランが実行またはスケジュール設定されるまで確立されません。確立されると、Design Studio はユーザーが終了するまで接続を保持します。

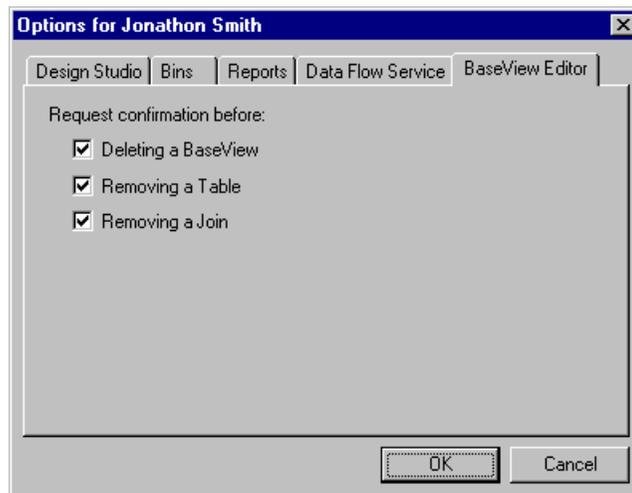
6. 「OK」をクリックします。

BaseView エディタの設定の変更

BaseView エディタでオブジェクトを削除する場合、Design Studio によって確認が求められます。BaseView、表および結合に対して確認プロンプトを設定できます。

BaseView エディタのプロンプトを変更するには、次の手順を実行します。

1. 「**BaseView Editor**」タブをクリックします。



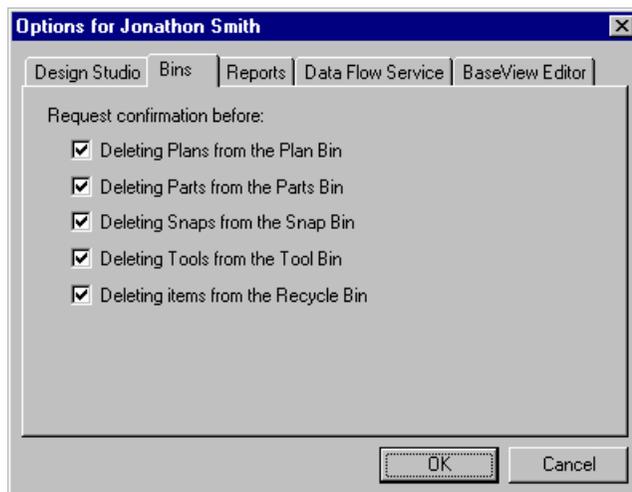
2. チェック・ボックスを選択または選択解除して、プロンプト・タイプを選択または選択解除します。
3. 「OK」をクリックします。

Bin の設定の変更

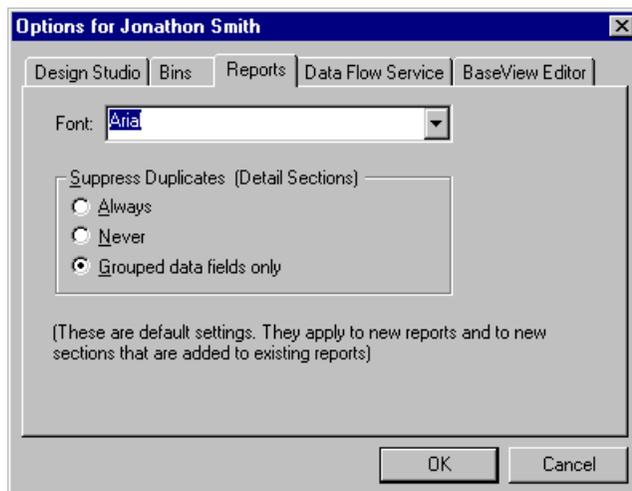
Bin のオブジェクトを削除する場合、Design Studio によって確認が求められます。Bin アイテムのタイプごとに確認プロンプトを設定できます。

Bin のプロンプトを変更するには、次の手順を実行します。

1. 「Bins」 タブをクリックします。



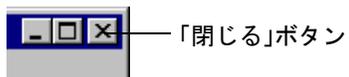
2. チェック・ボックスを選択または選択解除して、プロンプト・タイプを選択または選択解除します。
3. 「OK」 をクリックします。



Design Studio のクローズ

Design Studio を閉じるには、次のいずれかの手順を実行します。

- 画面の左上にある「**System**」 ボタンをダブルクリックします。
- 画面の右上にある「**閉じる**」 ボタンをクリックします。



- 「**File**」 → 「**Exit**」 を選択します。

BaseView の設計

ここで説明する内容は次のとおりです。

- BaseView の概要
- BaseView での作業
- BaseView の変更

BaseView の概要

BaseView は、表、列、結合など、データベースのすべてまたは一部のグラフィカルな表現です。BaseView エディタには、この情報が図解で表示され、BaseView を編集できます。BaseView を作成する場合は、データベースと有効なログインを指定します。Design Studio によって、参照するデータ構造とデータ・ディクショナリについて記述されているデータベース・カタログが読み取られます。次に、BaseView に追加する表をデータベースから選択します。後で、データベースから他の表を追加できます。

BaseView の作成後に、その BaseView を使用する MetaView を作成できます。MetaView には、プランおよび Snap を作成する Design Studio ユーザーがアクセスします。ユーザーは、Parts Bin 内の Category および Part として MetaView の内容を表示します。

同じデータベースから個別の BaseView を作成できます。Oracle BAM Enterprise Link 管理者用のビューにはシステム表を含め、ユーザー用のビューにはデータ表のみを含めるような場合があります。

すべての MetaView は、BaseView によって表現される基礎となる構造に基づき、表結合またはリレーションシップは、BaseView に定義されます。このため、すべての MetaView で発生するような変更の場合は、BaseView でその変更を行います。

BaseView に対して変更を行っても、基礎となるデータベースは変更されません。

BaseView のセキュリティは、Oracle BAM Enterprise Link Admin で設定します。詳細は、『Oracle Business Activity Monitoring Enterprise Link 管理者ガイド』を参照してください。

BaseView での作業

BaseView エディタを使用して、新しい BaseView を作成します。

- [BaseView の作成](#)
- [Snap 用の BaseView の作成](#)
- [BaseView エディタの使用](#)

BaseView の作成

新しい BaseView を作成するには、使用しているデータベース・タイプへの接続に必要なパラメータを知っておく必要があります。多くのデータベース・タイプでは、次の項目を知っておく必要があります。

- データベース名
- サーバー名
- データベースのログインとパスワード

BaseView を作成するには、次の手順を実行します。

1. 「Tools」 → 「BaseView Editor」 を選択します。
2. BaseView ツールバーで 「New BaseView」 ボタンをクリックし、「Define BaseView」 ダイアログ・ボックスを表示します。



「New BaseView」ボタン

3. 次の情報を入力します。
 - BaseView Name: BaseView に割り当てる名前。この名前は、現行のリポジトリ内で一意である必要があります。
 - Database Type: 選択するデータベースのタイプ。
 - Database Dictionary Type: この BaseView で使用するメタデータのソース。選択するタイプに応じて、次の項目を参照してください。

データ・ディクショナリのタイプ	説明
Microsoft Repository	32 ページの「Microsoft Repository を使用した BaseView の作成」 を参照。
iWay Connector	33 ページの「iWay Connector を使用した BaseView の作成」

- Server: データベースが配置されているサーバーの名前。
- Database: BaseView の作成元となるデータベースの名前。
- User Id: BaseView データベースにアクセスするデータベース・ユーザーの名前。このユーザー ID は、有効なデータベース・ログインである必要があります。
- Password: データベース・ユーザーのパスワード。
- Database User: ユーザー ID がマップされるデータベース・ユーザーの名前。たとえば、Microsoft SQL Server では、ユーザー ID を dbo にマップして、ユーザーにすべての権限を付与する場合があります。

「SQL Server」を選択した場合、接続タイプのラジオ・ボタンが有効になります。「Use Trusted Connection」を選択すると、「User Id」、「Password」および「Database User」フィールドが無効になります。信頼できる接続とは、デフォルトのユーザー情報を使用してデータベースにログインすることを意味します。このデフォルトのログインは、Oracle BAM Enterprise Link Admin で設定します。これ以外の場合は、標準の接続を使用します。

フィールドの横にある赤色の矢印は、フィールドが必須であるか、または値が無効であることを示しています。たとえば、「BaseView Name」フィールドに既存の BaseView の名前を入力すると、赤色の矢印でマーク付けされます。「Database User」フィールドは、「User Id」フィールドにリンクされています。「User Id」に入力した値は、「Database User」に適用されます。

「Server」フィールドのラベルは、「Database Type」で選択した値によって異なります。

- 「Database Type」で「Oracle」を選択した場合、「Server」フィールドは「Host String」に変わります。この文字列は、<domain.instancename> という書式になります。
- 「Database Type」で「ODBC」を選択した場合、「Server」フィールドは「Data Source」に変わります。これは、ODBC 管理アプリケーションで作成したデータ・ソースの名前です。Data Flow Service マシンのデータ・ソース名と完全に一致させる必要があります。

Microsoft SQL Server の BaseView の場合、「Define BaseView」ダイアログ・ボックスは次のようになります。

Oracle の BaseView の場合、このダイアログ・ボックスは次のようになります。

Oracle の BaseView の場合、「BaseView Name」および「User Id」の値には、すべて大文字を使用します。

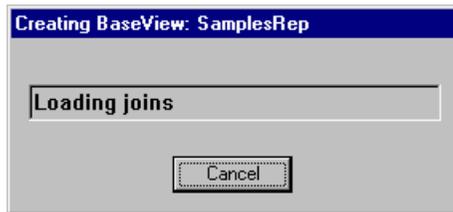
4. BaseView に追加する表を選択する場合は、「Tables」をクリックします。

デフォルトでは、すべてのユーザー表が選択されます。すべての表を追加するか、またはリストから一部の表を選択できます。これは、多くの表を含むデータベースの一部を追加する場合に役立ちます。このダイアログ・ボックスで指定した接続情報は、使用可能な表を表示するために完全で正確である必要があります。

BaseView には、異なるレベルのユーザーのデータベース情報を含めることができます。Oracle BAM Enterprise Link 管理者はシステム表を含む BaseView を所有でき、データ設計者はビューおよびシノニムを含む BaseView を所有できます。

5. 表を物理名ではなく表示名で表示するには、「**Show display names**」をクリックします。詳細は、41 ページの「**物理名または表示名の表示**」を参照してください。
6. BaseView の MetaView を自動的に作成する場合は、「**Autocreate MetaView**」チェック・ボックスを選択して、MetaView の名前を入力します。
7. 「**OK**」をクリックして、BaseView を追加します。

BaseView エディタによって、BaseView の作成に応じた進捗メッセージが表示されます。BaseView が作成中の場合は、「**Cancel**」をクリックして作成を取り消すことができます。



信頼できるセキュリティを持つ BaseView の作成

Design Studio では、Microsoft SQL Server の BaseView における信頼できるセキュリティのサポートが提供されます。信頼できるセキュリティを使用すると、統合または混在したセキュリティ構成でサーバーにアクセスできます。個々の SQL Server ログインは必要ないため、データ・マート・アクセスの設定および管理が容易になります。

Microsoft SQL Server の BaseView に対して信頼できるセキュリティを実装するには、構成が次の内容に準拠している必要があります。

- SQL Server および Data Flow Service は、同じサーバー・マシンで動作している必要があります。
- Data Flow Service は、名前付きパイプを使用して SQL Server と通信する必要があります。
- クライアント・マシンは、名前付きパイプを使用して Data Flow Service と通信する必要があります。
- クライアント・マシンは、名前付きパイプを使用して SQL Server と通信する必要があります。

名前付きパイプを使用して SQL Server と通信するようにクライアント・マシンを構成するには、次の手順を実行します。

1. SQL Server クライアント設定ユーティリティを開きます。
2. 「拡張」タブをクリックし、次の項目を指定します。
 - サーバー : SQL Server 名
 - DLL 名 : Named Pipes
 - コネクション : ¥¥.¥PIPE¥SQL¥QUERY
3. 「追加」をクリックします。これで、接続がローカルの名前付きパイプとして作成されます。

BaseView に対して信頼できるセキュリティを実装するには、次の手順を実行します。

1. 信頼できるデータベース・ログインを使用して BaseView を作成します。
2. Oracle BAM Enterprise Link Admin で、データにアクセスするユーザーのために、信頼できる BaseView ログインを定義およびマップします。

「Define BaseView」ダイアログ・ボックスに、信頼できる BaseView を作成するためのオプションが表示されます。

The image shows the 'Define BaseView' dialog box. It has a title bar 'Define BaseView'. The fields are: BaseView Name: 'Untitled1', Database Type: 'SQL SERVER', Data Dictionary Type: 'DATABASE', Server: (empty), Database: (empty). There are two red arrows pointing to the Server and Database fields. Below these are two radio buttons: 'Use Standard Connection' (unselected) and 'Use Trusted Connection' (selected). Under 'Use Standard Connection' are fields for User Id, Password, and Database User. Under 'Use Trusted Connection' are checkboxes for 'Snap BaseView' and 'Auto Create MetaView'. At the bottom are buttons for 'OK', 'Cancel', and 'Tables >>'.

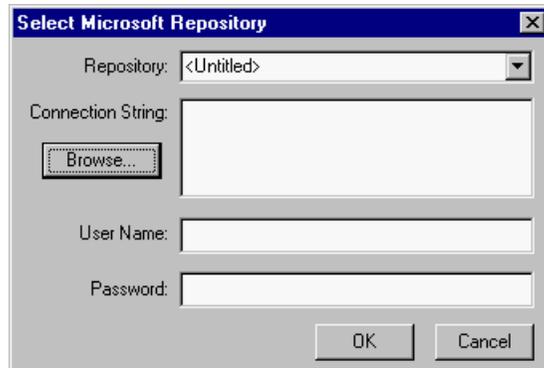
Oracle BAM Enterprise Link Admin では、BaseView ログインを追加するためのダイアログ・ボックスに、信頼できるログインを作成するためのオプションも表示されます。TrustedSecuritySupported 設定は、BaseView が信頼できるログインで作成できるかどうかを示します。

注意： SQL Server の場合、そのデフォルト値は「Yes」です。この設定が「No」である場合、ダイアログ・ボックスに信頼できるログイン用のオプションは表示されません。

Microsoft Repository を使用した BaseView の作成

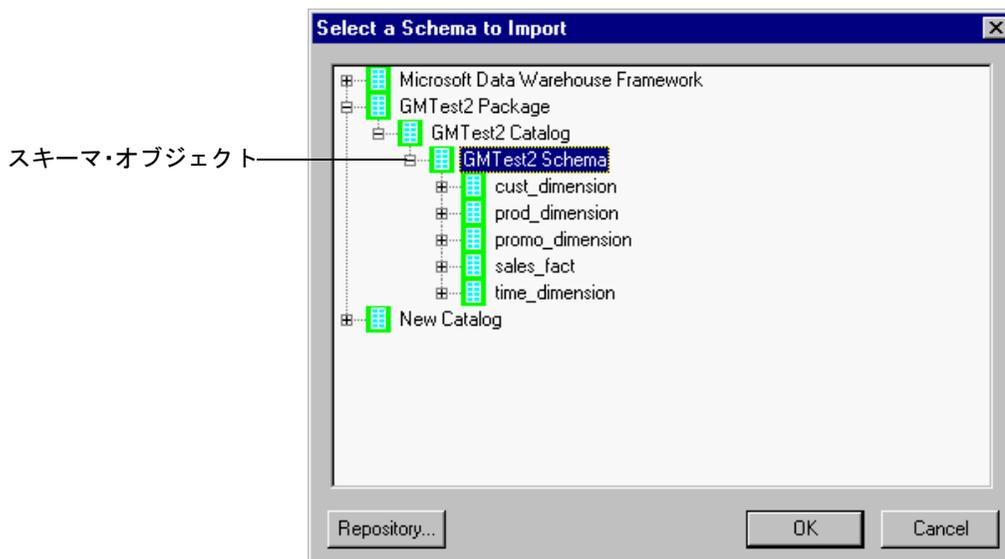
Microsoft Repository データ・ディクショナリを使用して BaseView を作成するには、次の手順を実行します。

1. 27 ページの「BaseView の作成」の手順に従います。
2. 「Data Dictionary Type」ドロップダウン・リストから「**Microsoft Repository**」を選択します。
「Select Microsoft Repository」ダイアログ・ボックスが表示されます。



3. Microsoft Repository のログイン情報を入力して、「**OK**」をクリックします。
 - **Repository:** Microsoft Repository の名前。
 - **Connection String:** Microsoft Repository の接続文字列。データベースが SQL Server の場合は、`server=servername;database=msdb` という書式を使用します。データベースが Microsoft Access ファイルの場合は、「**Browse**」ボタンをクリックし、そのファイルの場所を特定して、`C:¥access.mdb` という書式を使用します。
 - **User Name:** データベース・ユーザーの名前。
 - **Password:** ユーザー・アカウントのパスワード。

「Select a Schema to Import」ダイアログ・ボックスが表示され、Microsoft Repository オブジェクトの階層が示されます。



- ツリー・コントロールからスキーマ・オブジェクトを選択し、「OK」をクリックして「Create BaseView」ダイアログ・ボックスに戻ります。スキーマ・オブジェクトは、BaseView のレベルです。リポジトリを切り替える必要がある場合は、「Repository」ボタンをクリックして、「Select Microsoft Repository」ダイアログ・ボックスを表示できます。

Microsoft Repository との統合の詳細は、167 ページの「[Microsoft Repository での作業](#)」を参照してください。

iWay Connector を使用した BaseView の作成

iWay Connector データ・ディクショナリ・タイプを使用して BaseView を作成するには、次の手順を実行します。

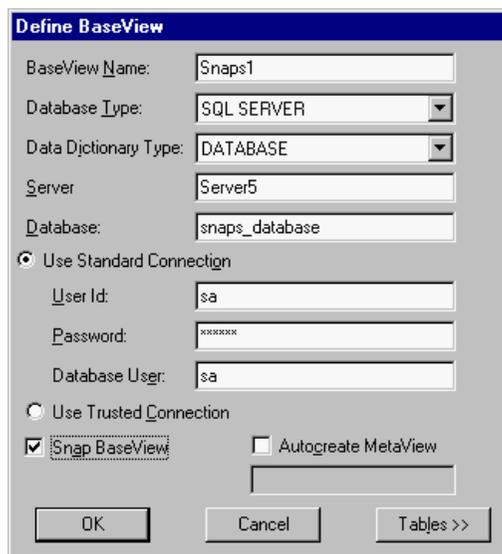
- 27 ページの「[BaseView の作成](#)」の手順に従います。
- 「Data Dictionary Type」ドロップダウン・リストから「iWay Connector」を選択します。

Snap 用の BaseView の作成

Snap は、リレーショナル・データベース表として保存されます。Snap が保存されるデータベースを指す BaseView を作成する必要があります。この BaseView を作成する場合、Snap 表の所有者として使用するデータベース・ユーザーを指定します。このデータベース・ユーザーには、指定するデータベースで表を作成する権限があることを確認します。

Snap 用の BaseView を作成するには、次の手順を実行します。

1. 「Tools」 → 「BaseView Editor」 を選択します。
2. BaseView ツールバーで 「New BaseView」 ボタンをクリックし、「Define BaseView」 ダイアログ・ボックスを表示します。



The image shows the 'Define BaseView' dialog box. It has a title bar 'Define BaseView'. The fields are: BaseView Name: Snaps1; Database Type: SQL SERVER; Data Dictionary Type: DATABASE; Server: Server5; Database: snaps_database. There are two radio buttons: 'Use Standard Connection' (selected) and 'Use Trusted Connection'. Under 'Use Standard Connection', there are fields for User Id: sa, Password: xxxxxxxx, and Database User: sa. There are two checkboxes: 'Snap BaseView' (checked) and 'Autogreate MetaView' (unchecked). At the bottom, there are buttons for 'OK', 'Cancel', and 'Tables >>'.

3. データベースについて、必須情報を入力します。
4. 「Snap BaseView」 チェック・ボックスを選択します。

BaseView に追加する表を選択する場合は、「Tables」をクリックします。データベースに表がすでに含まれている場合、これらの表を BaseView に表示させないようにすることができます。「User」表タイプ・チェック・ボックスを選択して、表示させない表を選択解除します。

5. BaseView の MetaView を自動的に作成する場合は、「Autocreate MetaView」 チェック・ボックスを選択して、MetaView の名前を入力します。

Snap 用の MetaView を使用すると、Snap 表に対する問合せを直接実行できます。

6. 「OK」 をクリックして、BaseView を追加します。

BaseView が作成され、Snap のコンテナとして指定されます。Oracle BAM Enterprise Link 管理者は、BaseView および MetaView のセキュリティと、Snap 表の所有権およびアクセスを設定します。詳細は、『Oracle Business Activity Monitoring Enterprise Link 管理者ガイド』を参照してください。

ユーザーが Snap を保存すると、Snap 表がデータベース、BaseView および MetaView に自動的に追加されます。

BaseView エディタの使用

新しい BaseView を作成する際に、Design Studio では、BaseView エディタに物理データベースの表および結合が表示されます。BaseView エディタで、表の移動とサイズ変更、および BaseView の削除を実行できます。

ここで説明する内容は次のとおりです。

- [BaseView の選択](#)
- [BaseView 内の表の表示](#)
- [表の移動](#)
- [BaseView のコピー](#)
- [BaseView の削除](#)

BaseView の選択

BaseView エディタには、現行の BaseView が表示されます。リポジトリ内の他の BaseView は、「BaseView」ドロップダウン・リストに表示されます。

BaseView を選択するには、次の手順を実行します。

1. 「Tools」→「BaseView Editor」を選択します。

BaseView エディタには、最後に選択された BaseView が表示されます。

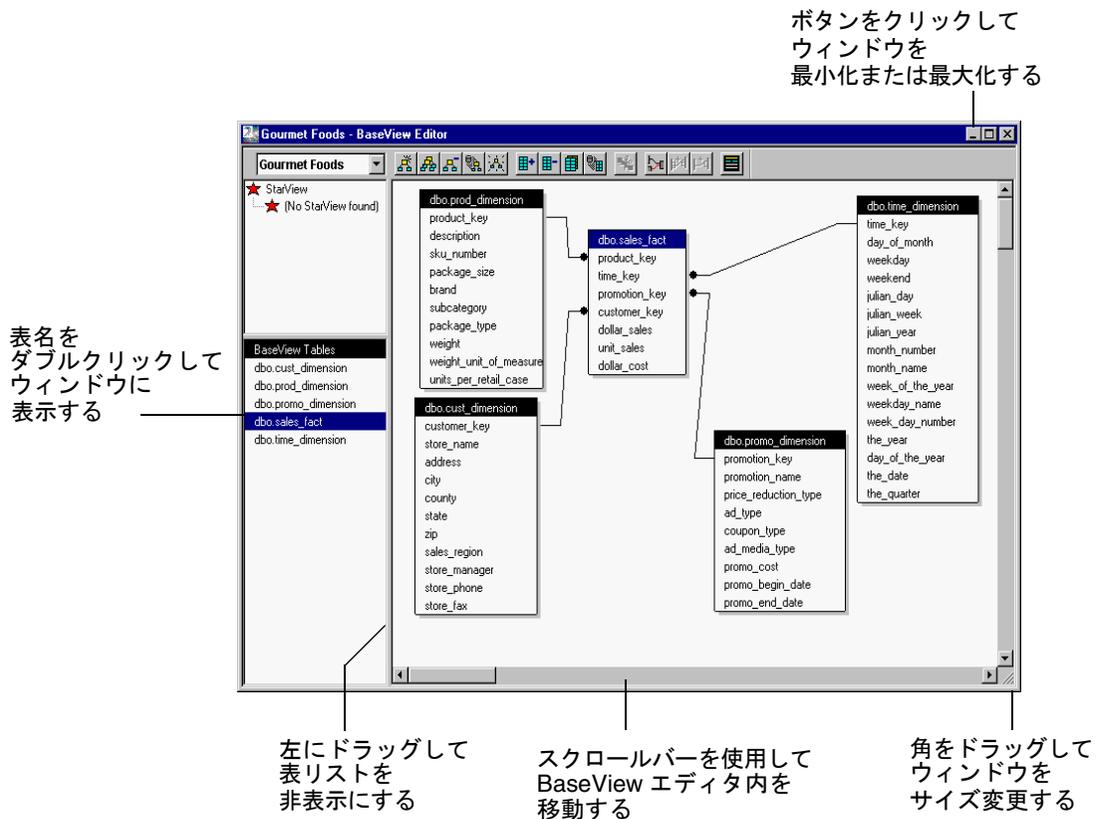
2. BaseView エディタの上部にある「BaseView」ドロップダウン・リストで、BaseView の名前を選択します。

BaseView エディタには、選択した BaseView の表が表示されます。

BaseView 内の表の表示

BaseView エディタ・ウィンドウには、現在表示されていない表に移動するために、縦方向と横方向のスクロールバーがあります。BaseView エディタに部分的に表示されている表を表示するには、ウィンドウの任意の角をドラッグしてサイズ変更します。また、Design Studio の他の領域で作業している間、ウィンドウを最小化することもできます。

BaseView 内の表リストは、表ロケータとして機能します。BaseView エディタを開くと、表リストがデフォルトで表示されます。表名をダブルクリックして、BaseView エディタにその表を表示します。



表リストから Parts Bin に表をドラッグすると、Category を MetaView に追加できます。

表リストを非表示にするには、BaseView エディタ・ウィンドウの左側の縁のハンドルにカーソルを置き、左にドラッグします。

表の移動

BaseView エディタで表を再配置する場合：

- 表のタイトル・バーをクリックして表を選択し、新しい場所にドラッグします。

表をサイズ変更する場合：

- 表の縁（カーソルがサイズ・ハンドルに変化する箇所）を選択し、ドラッグして表をサイズ変更します。

BaseView のコピー

新しい BaseView は、既存の BaseView をコピーして作成できます。BaseView のコピーを作成すると、BaseView の表、別名、列および結合がコピーされます。StarView はコピーされません。StarView の詳細は、[131 ページの「StarMarts の設計」](#)を参照してください。

BaseView をコピーするには、次の手順を実行します。

1. 「BaseView」ドロップダウン・リストからコピーする BaseView を選択し、「Copy BaseView」ボタンをクリックします。



「Copy BaseView」ボタン

「Copy BaseView」ダイアログ・ボックスが表示されます。

2. 新しい BaseView の名前を入力し、「OK」をクリックします。

BaseView エディタに新しい BaseView が表示されます。

BaseView の削除

BaseView エディタで BaseView を削除する場合は、リポジトリから BaseView の定義を削除します。これによって、基礎となるデータベースに影響することはありません。

注意： BaseView を削除する前に、その BaseView で生成された MetaView から Category および Part を削除する必要があります。削除された BaseView の Part を含むプランは実行できません。Part のプロパティには、その Part の BaseView が表示されます。詳細は、[68 ページの「Part のプロパティの表示」](#)を参照してください。

BaseView を削除するには、次の手順を実行します。

1. 「BaseView」ドロップダウン・リストで BaseView を選択し、ツールバーで「Delete BaseView」ボタンをクリックします。



「Delete BaseView」ボタン

メッセージによって、この BaseView の削除について確認が求められます。確認メッセージは、カスタマイズ可能なユーザー・オプションです。詳細は、[22 ページの「BaseView エディタの設定の変更」](#)を参照してください。

2. 「OK」をクリックします。

BaseView が削除されます。

BaseView の変更

BaseView エディタを使用して BaseView を変更します。実行可能な内容は次のとおりです。

- プロパティの変更
- 表の追加、名前の変更または削除
- 結合グループの作成
- 結合タイプの変更
- 別名表の作成
- 表、列および結合の再同期化

BaseView を編集して、MetaView の作成に必要な表、列および結合を指定します。

注意： BaseView に対するすべての変更は、BaseView エディタを閉じるか、またはクリックして他に移動すると自動的に保存されます。これらの変更が、基礎となるデータベースに影響することはありません。

ここで説明する内容は次のとおりです。

- [BaseView のプロパティの変更](#)
- [表のプロパティの表示](#)
- [列のプロパティの表示](#)
- [物理名または表示名の表示](#)
- [BaseView への表の追加](#)
- [表の削除](#)
- [BaseView の結合](#)
- [別名表の作成](#)
- [表の自己結合](#)
- [BaseView の再同期化](#)

BaseView のプロパティの変更

あるリポジトリに作成したオブジェクトを別の環境に移行しても、その新しいリポジトリで BaseView のプロパティを変更できます。データベースおよびサーバー名は、BaseView を再作成しなくても変更できます。

BaseView のプロパティを変更するには、次の手順を実行します。

1. BaseView エディタで BaseView を開きます。
2. 右クリックして、「**BaseView "Name" Properties**」を選択します。
「BaseView Properties」ダイアログ・ボックスが表示されます。
3. 必要に応じて、BaseView 名、サーバーおよびデータベースを編集します。
4. 「OK」をクリックしてダイアログ・ボックスを閉じます。

表のプロパティの表示

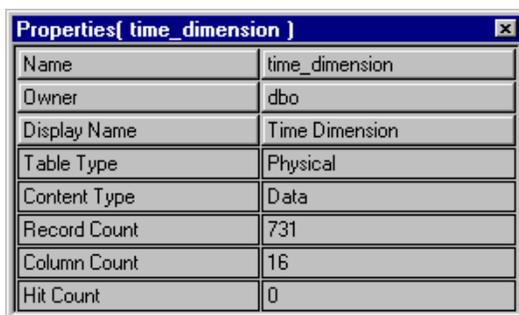
「Properties」ダイアログ・ボックスを使用して、選択した表についての情報を表示します。表名、所有者および表示名を編集できます。

表のプロパティを表示するには、次の手順を実行します。

- ▶ タイトル・バーをクリックして表を選択し、ツールバーで「**View Properties**」ボタンをクリックします。また、表のタイトル・バーをダブルクリックすることもできます。



「Properties」ダイアログ・ボックスに表のプロパティが表示されます。



「Name」、「Owner」および「Display Name」フィールドを編集できます。平坦に見える項目は、読取り専用です。プロパティを編集する場合は、BaseView 内のメタデータのみを変更します。基礎となるデータベースは変更しません。

プロパティ	説明
Name	データベース名と一致する表の物理名。物理名は、表のデフォルトの名前です。表示名と物理名の表示を切り替えることができます。
Owner	表のデータベース所有者。
Display Name	表の表示名。これは、編集するまで物理名と同じです。表示名を編集した場合は、MetaView の作成時にその表示名が使用されます。
Record Count	表内のレコード数。この値は、ユーザーが表を含むプランを実行した後に表示されます。
Content Type	データベースのコンテンツのタイプ: データ、ファクト、ファクトレス・ファクト、ディメンション、集計ファクト、集計ディメンション
Column Count	表内の列数。
Hit Count	表がクライアント・アプリケーションからアクセスされた回数。

「Table Properties」ダイアログ・ボックスで値を編集するには、次の手順を実行します。

➤ 項目をダブルクリックし、新しい値を入力して [Enter] キーを押します。

「Properties」ダイアログ・ボックスを開いたままにして、他の表または列を選択し、そのプロパティを表示できます。

注意： BaseView 内の表には、その基礎となるデータベース表と同じ物理名が必要です。BaseView の表名を変更するのは、データベース表の名前を変更した場合のみです。表示名は、MetaView を作成する目的で変更できます。

列のプロパティの表示

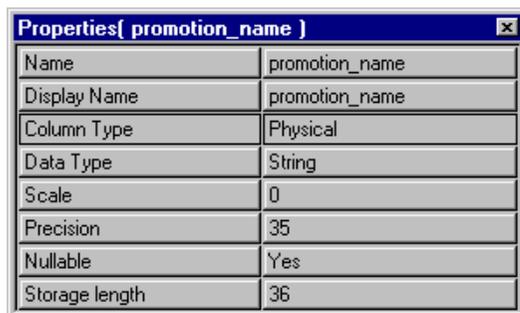
「Properties」ダイアログ・ボックスを使用して、BaseView 表の列についての情報を表示および編集します。基礎となるデータベース内の表が変更された場合は、列のプロパティの編集が役立ちます。BaseView 内のメタデータを更新するには、次のいずれかを実行できます。

- 表を削除してから、再び追加します。BaseView 表を削除すると、その表から作成した MetaView の Part が無効になります。
- BaseView を再同期化します。BaseView を再同期化すると、BaseView 内の表は更新されますが、その表から作成された MetaView の Part には影響しません。
- データベースと一致するように、BaseView 内の列のプロパティを編集します。列のプロパティを編集すると、その列に関連付けられている MetaView の Part が保存されます。

列のプロパティを表示するには、次の手順を実行します。

➤ 列を選択し、ツールバーで「View Properties」ボタンをクリックします。列を右クリックして、「Properties」を選択することもできます。

「Properties」ダイアログ・ボックスに列のプロパティが表示されます。



Properties[promotion_name]	
Name	promotion_name
Display Name	promotion_name
Column Type	Physical
Data Type	String
Scale	0
Precision	35
Nullable	Yes
Storage length	36

列のプロパティはいずれも編集できます。プロパティを編集する場合は、BaseView エディタでメタデータのみを変更します。基礎となるデータベースは変更しません。

プロパティ	説明
Name	BaseView 内の列の物理名。
Display Name	列の表示名。これは、MetaView で使用される名前です。
Column Type	「Physical」は、これが通常のデータベース列であることを示します。
Data Type	列の Enterprise Link データベース・タイプ。ほとんどの場合、データ型は、基礎となるデータベースのデータ型に一致します。
Scale	DECIMAL 列における列のスケール。
Precision	DECIMAL 列における列の精度。
Nullable	列で NULL 値を許可するかどうかを決定するフラグ。
Storage Length	列の格納長。

「Column Properties」ダイアログ・ボックスで値を編集するには、次の手順を実行します。

- 編集する項目をダブルクリックし、新しい値を入力して [Enter] キーを押します。

「Properties」ダイアログ・ボックスを開いたままにして、他の列または表を選択し、そのプロパティを表示できます。

注意： BaseView エディタでは、列のプロパティ値は検証されません。BaseView のプロパティ値は、データベース表の値と一致させる必要があります。

物理名または表示名の表示

BaseView 内の表と列には、物理名と表示名があります。表示名は、表または列のプロパティで編集できます。物理名は、基礎となるデータベースで使用されるオリジナルの名前です。表の物理名には、表所有者名の接頭辞がありません。MetaView は表示名を使用し、StarView は物理名を使用します。

BaseView 内の表示名を表示するには、次の手順を実行します。

- 右クリックして、「Show」→「Display Names」を選択します。

表および列が、表示名で表示されます。

物理名に切り替えるには、右クリックして、「Show」→「Physical Names」を選択します。

BaseView への表の追加

BaseView を作成した後、同じデータベースから他の表を追加できます。これは、表を徐々に追加する場合に役立ちます。また、BaseView と基礎となるデータベースを再同期化しても、BaseView に表を追加できます。詳細は、52 ページの「BaseView の再同期化」を参照してください。

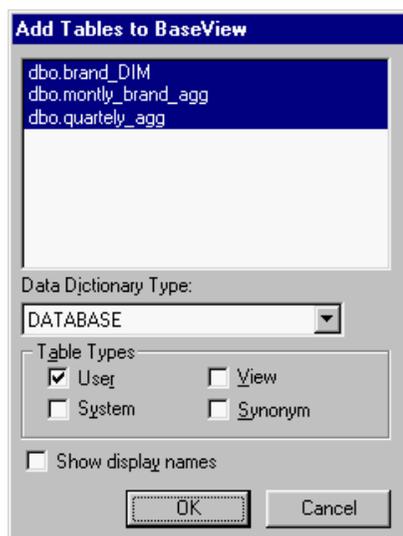
表を BaseView に追加するには、次の手順を実行します。

1. BaseView エディタで、「Add Table」ボタンをクリックして、「Add Tables to BaseView」ダイアログ・ボックスを表示します。



「Add Table」ボタン

ダイアログ・ボックスには、BaseView に追加されていないユーザー表が表示されます。



ユーザー表、システム表、ビューおよびシノニムを選択できます。

表示名で表を表示するには、「Show display names」を選択します。

別のタイプのデータベース・ディクショナリで記述された表を追加するには、ドロップダウン・リストから該当するデータ・ディクショナリ・タイプを選択します。使用している環境のログイン情報を入力して、「OK」をクリックします。

2. 追加する表を選択します。複数の表を選択する場合は、カーソルをドラッグするか、または [Ctrl] キーと [Shift] キーを使用します。
3. 「OK」をクリックして、表を追加します。

表の削除

BaseView から表を削除しても、データベースは変更されません。BaseView から表を削除すると、その表に基づく MetaView の Part が無効になります。表を変更する場合は、表を削除して再び追加するのではなく、BaseView を再同期化します。

BaseView に StarView が存在する場合に、その BaseView から表が削除されると、StarView から該当する表が削除されます。

BaseView から表を削除するには、次の手順を実行します。

1. BaseView エディタで表のタイトル・バーをクリックして表を選択し、ツールバーで「Remove Table」ボタンをクリックします。

表を右クリックして、「Remove Table」を選択することもできます。



「Remove Table」ボタン

2. メッセージによって、この表の削除について確認が求められます。「OK」をクリックして表を削除します。

BaseView から削除した表は、「Add Table」ボタンを使用して再び追加できます。

確認メッセージは、カスタマイズ可能なユーザー・オプションです。詳細は、[22 ページの「BaseView エディタの設定の変更」](#)を参照してください。

BaseView の結合

BaseView を作成すると、物理データベース内の結合がその BaseView に含まれます。BaseView エディタでは、追加の結合の作成、結合タイプの変更、および between 結合と連結結合を実行できます。BaseView で結合を追加または変更しても、基礎となるデータベースは変更されません。

ここで説明する内容は次のとおりです。

- [結合の追加](#)
- [結合の名前の変更](#)
- [結合の削除](#)
- [結合タイプの変更](#)
- [between 結合の作成](#)
- [連結結合の追加](#)
- [結合グループの定義](#)

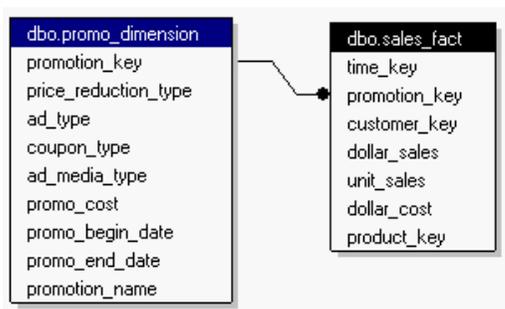
結合の追加

BaseView 内の表間における追加の結合を定義できます。プランでは、これらの結合を使用して、個別の表にある情報を関連付けます。結合される列は、同じ名前である必要はありませんが、同じデータ型の関連するデータが含まれている必要があります。結合を追加する前に、両方の列が表示されるように BaseView エディタで表を配置します。

表間の結合を追加するには、次の手順を実行します。

1. 結合する一方の列を選択します。
2. 結合する他方の列までドラッグします。

この例では、sales_fact 表と promo_dimension 表を promotion_key で結合します。



つながっている黒色の線は、結合を表しています。デフォルトでは、結合は、結合を作成するためにドラッグした方向へ 1 対多の関係を持つ内部結合です。スター・スキーマの場合、ディメンション表からファクト表にドラッグします。

BaseView エディタで結合にカーソルを置くと、結合されている列が強調表示されます。

結合の名前の変更

結合には、名前を割り当てることができます。デフォルトの結合名は、BaseView の 2 つの表名を組み合わせたものです。複雑な BaseView で、使用または編集する結合を特定するために結合名を利用できます。

結合の名前を変更するには、次の手順を実行します。

1. 右クリックして、「**Rename Join "Name"**」を選択します。
「Rename Join」ダイアログ・ボックスが表示されます。
2. 新しい結合名を入力し、「**OK**」をクリックします。

結合の削除

使用しない結合を削除して、問合せ環境をクリーンアップできます。集計表がある場合に二重計算を回避するために、これを行うことがあります。結合を削除しても、データベースは変更されません。BaseView 内の表間の結合をすべて削除すると、異なる表の列をプランで使用できなくなることに注意してください。

結合を削除するには、次の手順を実行します。

1. 削除する結合をクリックします。
選択された結合は青色になります。
2. ツールバーで「Delete Join」ボタンをクリックします。



「Delete Join」ボタン

3. メッセージによって、この結合の削除について確認が求められます。「OK」をクリックして結合を削除します。

連結結合を削除すると、BaseView エディタによって、連結キーを定義する結合がそれぞれ削除されます。

結合を削除しても、すでに保存されているプランには影響しません。これは、プランの問合せがその結合に基づいてすでに生成されているためです。

確認メッセージは、カスタマイズ可能なユーザー・オプションです。詳細は、[22 ページの「BaseView エディタの設定の変更」](#)を参照してください。

結合タイプの変更

BaseView で結合を作成すると、その結合は内部結合になります。内部結合には、両方の表に一致する列値がある行のみが含まれます。また、Design Studio では、左側外部結合および右側外部結合もサポートされます。外部結合を作成するには、既存の内部結合を変更します。

外部結合には、関連する表では一致がない、ある表からの行が含まれます。左側外部結合には、表 B では一致がない、表 A からの行が含まれます。右側外部結合には、表 A では一致がない、表 B からの行が含まれます。

BaseView エディタで内部結合を外部結合に変更するには、次の手順を実行します。

- 結合を右クリックして、「Join type」→「Left Outer」または「Right Outer」を選択します。

既存の結合のタイプを表示するには、結合を選択してから右クリックして、結合メニューを表示します。タイプの横にチェック・マークが表示されます。

次の仕様が BaseView 内の結合に適用されます。

- 完全外部結合は、基礎となるデータベースでサポートされている場合に許可されます。基礎となるデータベースで外部結合がサポートされていない場合は、右クリック・メニューにある外部結合オプションは無効になります。
- 連結結合の場合、すべての結合は同じタイプである必要があります。連結結合については、47 ページの「連結結合の追加」を参照してください。
- 異なるタイプの結合を連結すると、警告メッセージによって、続行するか、取り消すかが尋ねられます。続行すると、新しい連結結合は、元の結合のタイプのいずれかに設定されます。
- 複数の結合を選択している場合に結合タイプを変更すると、選択されているすべての結合が変更されます。

between 結合の作成

ある表の同じ列から別の表の 2 つの列への結合を 2 つ作成する場合、この 2 つの結合を組み合わせて単一の **between** 結合にできます。**between** 演算子によって、ある表での値が、別の表での値の範囲と比較されます。**between** 結合を構成する列のデータ型には、互換性が必要です。

between 結合がサポートされるのは、基礎となるデータベースで **between** 結合に標準の WHERE 句構文が使用されている場合のみです。

between 結合を作成するには、次の手順を実行します。

1. [Ctrl] キーを押しながら各結合を選択して、2 つの結合を選択します。結合は、ある表の 1 つの列から別の表の 2 つの列に作成される必要があります。ある表の 2 つの列から別の表の 1 つの列に作成された結合を使用して、**between** 結合を作成することはできません。
2. BaseView エディタのツールバーで「**Between Join**」ボタンをクリックします。



「Between Join」ボタン

2 つの結合を選択した状態で右クリックし、「**Join Operator**」 → 「**Between**」を選択することもできます。

between 結合の外観は、連結結合と同じです。選択した結合が **between** 結合であれば、「**Between Join**」ツールバーがインデントされ、右クリックによる結合メニューでは「**Between**」が選択されます。

次の仕様が BaseView 内の **between** 結合に適用されます。

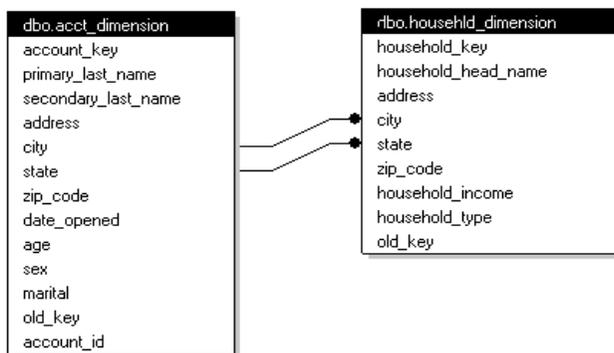
- **between** 結合は連結できません。連結結合は **between** 結合の構成に使用できません。
- **between** 結合は内部結合にのみすることができます。

連結結合の追加

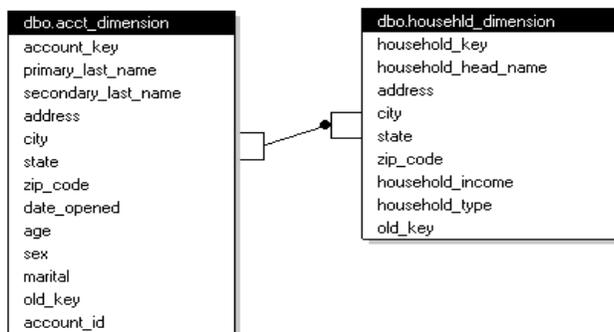
2つの表間の関係が列と列の結合より複雑な場合は、連結キーを使用します。BaseView で連結結合を定義して、連結キーを結合します。これによって、データベースが変更されることはありません。連結結合を作成するには、ある表で連結キーを構成する列から、別の表のキー列に結合を追加します。その後で、結合を選択し、それらを連結します。データベースで定義されている連結結合は、BaseView に表示されます。

連結結合を追加するには、次の手順を実行します。

1. 連結キーの列を1つ選択し、他の表の一致する列までドラッグします。
2. この連結キーを構成する他の列に対し、同じ操作を繰り返します。



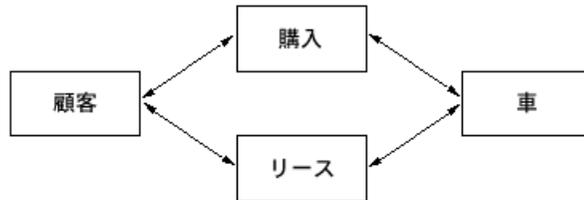
3. [Ctrl] キーを押しながら連結キーを定義する各結合を選択して、結合を選択します。他の結合は選択しないでください。
4. ツールバーで「Concatenate Joins」 ボタンをクリックします。複数の結合が、カッコ付きの単一の結合に変化します。



結合グループの定義

結合グループは、名前付きの結合または **MetaView** 用の結合の集合です。結合グループは、複数の表にある関連列をどのように結合できるかを定義します。結合グループは、プランで使用される列に複数の結合が存在する場合に、使用する結合パスを明確にします。

たとえば、車の販売特約店における取引について考えてみます。ある表には顧客情報が含まれ、別の表には取引に関係した車の情報が含まれています。購入の取引とリースの取引は、別々の表に記録されています。顧客と車の表は、購入とリースの両方の表を介して結合されています。



顧客と車を選択する問合せでは、これらの表は、リース表または購入表（あるいはその両方）を介して結合できます。必要な情報を取得するために最も効果的なパスを指定する必要があります。

スター・スキーマで構成されたデータベースから作成された **BaseView** では、追加の結合グループを定義する必要はありません。OLTP に使用される正規化されたデータベースでは、複数のパスでデータのセットを結合できます。

BaseView で結合グループを定義する場合は、「Join Groups」ダイアログ・ボックスで **MetaView** を選択します。**MetaView** には、同じ **BaseView** のための異なる結合グループを含めることができます。

結合グループを定義するには、次の手順を実行します。

1. ツールバーで「Join Groups」ボタンをクリックします。



「Join Groups」ボタン

「Edit Join Groups」ダイアログ・ボックスに、現行の MetaView における既存の結合グループが表示されます。



2. 結合グループを作成する MetaView を選択します。
3. BaseView エディタで、1 つ以上の結合をクリックして、結合グループに含めます。
選択した結合が、青色で表示されます。[Ctrl] キーを押しながら行をクリックして、選択または選択解除を行います。
4. 「Create New」ボタンの横のフィールドに、エンド・ユーザーが結合グループを選択する際に使用できるグループ名を入力します。
5. 「Create New」をクリックして、新しい結合グループをリストに追加します。
6. 「Close」をクリックします。

「Edit Join Groups」ダイアログ・ボックスで結合グループ名を選択して、結合グループを構成する結合を選択します。これらの結合は、BaseView エディタに青色で表示されます。

結合グループを削除するには、次の手順を実行します。

- リストで結合グループを選択して、「Remove」をクリックします。

これによって、結合が削除されることはありません。リストから結合グループが削除されるだけです。

結合グループを編集するには、次の手順を実行します。

1. リストで結合グループを選択します。
2. [Ctrl] キーを使用して、その結合グループの新しい結合のセットを選択します。
3. 「Replace」をクリックします。

結合グループの名前を変更するには、次の手順を実行します。

1. リストで結合グループを選択します。
2. 「**Properties**」 ボタンをクリックします。
「Join Group Properties」 ダイアログ・ボックスが表示されます。
3. 結合グループの新しい名前を入力します。
4. 「**OK**」 をクリックします。

別名表の作成

別名表は仮想的な表です。別名表は個別の表として扱うことができますが、これらの表は別名化しているソース表を指しているだけです。ソース表のコピーとは異なり、別名表は更新する必要がありません。これは、常にソース表の内容を反映します。

別名表を作成する理由として、次のいずれかが考えられます。

- ある表をその表自身と結合させるため
同じ表の複数のフィールドに対して条件を比較または配置する場合、その表とその表自身を結合できます。これは自己結合です。
- BaseView におけるループを解決するため
BaseView 内の 2 つの表間に複数のパスがある場合は、ループが存在します。ループによって、予期しない問合せ結果が発生する可能性があります。

BaseView に別名表を作成するには、次の手順を実行します。

1. BaseView エディタで表のタイトル・バーをクリックして表を選択します。
2. ツールバーで「**Create Alias Table**」 ボタンをクリックします。



「Create Alias Table」ボタン

「Assign Alias Name」 ダイアログ・ボックスが表示されます。

3. 「New Table Name」フィールドに別名表の名前を入力します。この名前は、BaseView 内で一意である必要があります。
4. 「**OK**」 をクリックします。

別名表が BaseView に追加されます。この時点で、この別名表と BaseView 内の他の表を結合し、この別名表の列に基づく MetaView の Part を追加できます。

表の自己結合

同じ表の複数の列に対して条件を比較または配置する場合、その表とその表自身を結合できます。このタイプの結合は自己結合と呼ばれ、別名表が必要になります。次の例では、必要な情報を取得する際に別名表がどのように役立つかを示します。

この例では、BaseView 内の次の表から情報を取得します。

dbo.employee
emp_id
emp_name
dob
ssn
mgr_id
region_id
address_id
salary_grade_id

`mgr_id` は、マネージャの `emp_id` です。次に示すような、従業員とそのマネージャが一致するプランを実行するには、BaseView で `mgr_id` 列と `emp_id` 列を結合する必要があります。

従業員名	社会保障番号	マネージャ名
John Smith	555-55-5555	Claire Johnson
Peter Barnes	444-44-4444	Claire Johnson
Claire Johnson	777-77-7777	Valerie Davies

`emp_id` と `mgr_id` の両方は Employee 表にあるため、Employee の別名表を作成し、BaseView で 2 つの列を結合します。

自己結合を作成した後で、従業員名とマネージャ名の Part を MetaView に追加できます。1 つの Part は Employee 表から、別の Part は Employee の別名表からのものになります。自己結合を使用すると、ユーザーはこれら 2 つの Part を結合して同じ問合せにすることができます。

BaseView の再同期化

データベース内の表が変更されても、BaseView ではその変更は自動的に反映されません。変更を BaseView に適用するには、BaseView と基礎となるデータベースを再同期化します。これは、BaseView または BaseView 内の表のサブセットを更新する場合に役立ちます。

BaseView エディタには、2つの再同期化オプションがあります。

- Resynchronize Entire BaseView

このオプションは、BaseView 内のすべてのアイテムに影響します。BaseView のリストからデータベース表を選択します。次に、表を追加または削除する設定と、BaseView およびデータベースを一致させる結合を指定します。

- Resynchronize BaseView Tables

このオプションは、表のサブセットと BaseView 内の結合を再同期化します。影響するのは、BaseView 内の選択された表のみです。

ここで説明する内容は次のとおりです。

- [BaseView 全体の再同期化](#)
- [BaseView 表の再同期化](#)
- [BaseView の結合の再同期化](#)

BaseView 全体の再同期化

BaseView を完全にリフレッシュし、その内容が基礎となるデータベースと一致していることを検証する場合は、BaseView 全体を再同期化します。BaseView 全体を再同期化するには、新しい表または変更された表のみでなく、BaseView で必要なデータベース表をすべて選択します。

BaseView 全体を再同期化するには、次の手順を実行します。

1. BaseView エディタのツールバーで「Resynchronize Entire BaseView」ボタンをクリックします。



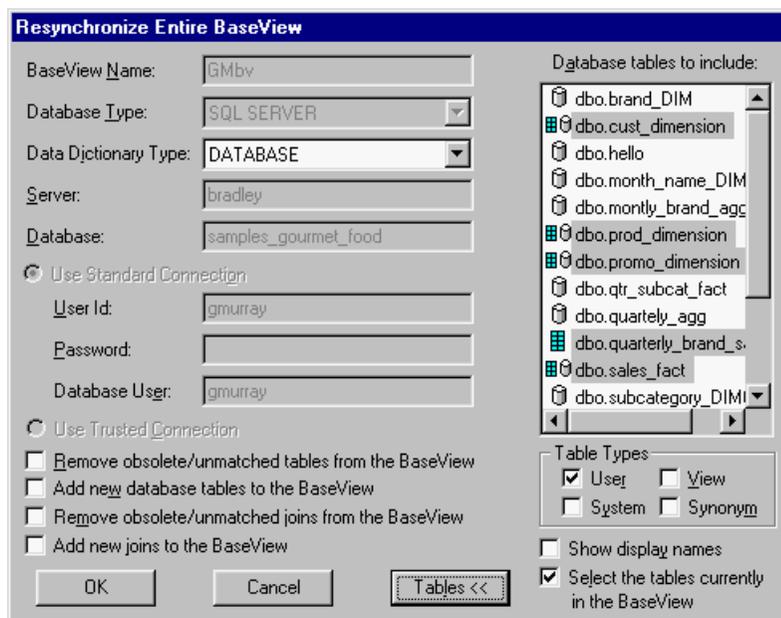
「Resynchronize Entire BaseView」ボタン

「Resynchronize Entire BaseView」ダイアログ・ボックスが表示されます。このダイアログ・ボックスでは、データベースへの接続に必要な情報は編集できません。この情報を編集する場合は、[38 ページの「BaseView のプロパティの変更」](#)を参照してください。

- 別のタイプのデータベース・ディクショナリで記述された BaseView 表を再同期化するには、ドロップダウン・リストから該当するデータ・ディクショナリ・タイプを選択します。使用している環境のログイン情報を入力して、「OK」をクリックします。

- 再同期化に含まれるデータベース表のリストを表示または変更するには、「Tables」ボタンをクリックします。

デフォルトでは、BaseView 内のすべての表が再同期化に含まれます。



表のリストでは、データベース表が BaseView に存在するかどうか、アイコンによって示されます。表のアイコンは BaseView を表し、円柱のアイコンはデータベースを表します。

- 他のタイプの表を含めるには、「Table Types」セクションで該当するチェック・ボックスを選択します。
- 表示名で表を表示するには、「Show display names」を選択します。
- 表のリストから、BaseView に含める表を選択します。複数の表を選択するには、[Shift] および [Ctrl] キーを使用します。

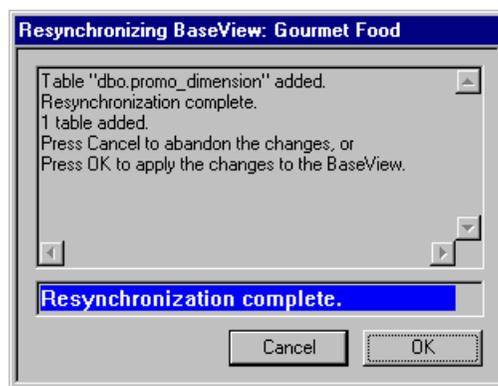
新しい表または変更された表のみでなく、BaseView で必要なすべての表を選択します。特定のタイプの表（システム表など）をすべて選択するには、「Select the tables currently in the BaseView」チェック・ボックスの選択を解除します。また、リストで表を個々にクリックして選択を解除することもできます。

7. BaseView を同期化する方法を 1 つ以上選択します。
 - データベースに存在しない表を BaseView から削除する場合は、「**Remove obsolete/unmatched tables from the BaseView**」を選択します。これによって、表のリストで選択されていない表が削除されます。
 - BaseView に存在しない表をデータベースから追加する場合は、「**Add new database tables to the BaseView**」を選択します。
 - データベースに存在しない結合を BaseView から削除する場合は、「**Remove obsolete/unmatched joins from the BaseView**」を選択します。
 - データベースから BaseView に結合を追加する場合は、「**Add new joins to the BaseView**」を選択します。

注意 :「**Remove obsolete/unmatched tables from the BaseView**」を選択すると、リストで選択されていない BaseView 内の表は、再同期化中に BaseView から削除されます。

8. 「OK」をクリックします。

BaseView エディタによって、BaseView と選択されたデータベース表が同期化され、ダイアログ・ボックスに結果が表示されます。



「Cancel」をクリックすると、変更を破棄できます。

9. 変更を BaseView に適用するには、「OK」をクリックします。

BaseView は、指定された変更を使用して更新されます。

BaseView 表の再同期化

BaseView 内の複数の表または結合を更新するには、「Resynchronize Table」機能を使用します。表を再同期化すると、その表に基づくすべての別名表も再同期化されます。BaseView 内の表を更新している場合は、「Add Table」または「Remove Table」ボタンを使用したり、「Column Properties」ダイアログ・ボックスで更新される列の名前を変更できます。

注意： BaseView から表を削除すると、その表から作成された MetaView の Part が無効になります。

データベースの表を削除または変更した後で BaseView 内の表を同期化するには、次の手順を実行します。

1. BaseView の列に対し、データベースで名前が変更された列の名前を変更します。

これによって、名前が変更された BaseView の列から作成された MetaView の Part と、オリジナルの Part を含むプランが、自動的に新しい列名を参照することを保証されます。

2. BaseView 内の表を選択し、データベース表と同期化します。複数の表を選択するには、[Ctrl] キーを使用します。

3. 右クリックして、「Resynchronize Table」を選択します。複数の表を選択している場合は、[Ctrl] キーを押しながら右クリックします。

選択した表が別名表であるか、または別名表のソースである場合は、右クリックして、「Resynchronize Source Table and Related Alias Tables」を選択します。

また、「Resynchronize Table」ボタンをクリックすることもできます。



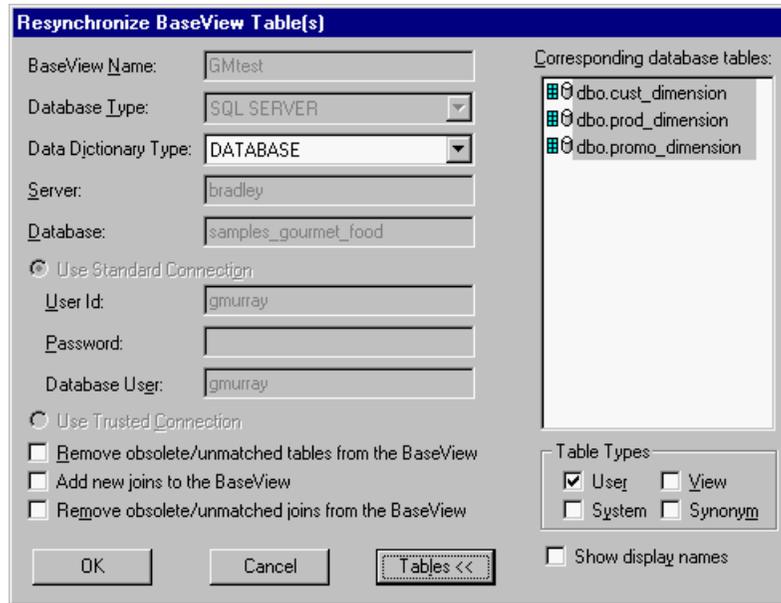
「Resynchronize Table」ボタン

「Resynchronize BaseView Tables」ダイアログ・ボックスが表示されます。このダイアログ・ボックスでは、データベースへの接続に必要な情報は編集できません。

4. 別のタイプのデータベース・ディクショナリで記述された表を含めるには、ドロップダウン・リストから該当するデータ・ディクショナリ・タイプを選択します。使用している環境のログイン情報を入力して、「OK」をクリックします。

- BaseView で選択されているデータベースの表のリストを表示するには、「Tables」ボタンをクリックします。

表のリストには、表が BaseView またはデータベース（あるいはその両方）に存在するかどうか、アイコンによって示されます。表のアイコンは BaseView を表し、円柱のアイコンはデータベースを表します。



BaseView の別名表を選択した場合、表のリストには選択した別名表のソース表が含まれます。選択した別名表は、表のリストには含まれません。

- ユーザー表以外のタイプの表を再同期化するには、「Table Types」セクションで該当するチェック・ボックスを選択します。
- 表示名で表を表示するには、「Show display names」を選択します。
- BaseView で削除する表をデータベースから削除している場合は、「Remove obsolete/unmatched tables from the BaseView」を選択します。

BaseView に StarView が存在する場合に、その BaseView から表が削除されると、StarView からも該当する表が削除されます。

- 「OK」をクリックします。

BaseView エディタによって、BaseView の選択された表が同期化され、ダイアログ・ボックスに結果が表示されます。BaseView 表の列およびプロパティは、データベースの表と一致するように変更されます。

- 「OK」をクリックして変更を BaseView に適用するか、または「Cancel」をクリックして変更を破棄します。

BaseView の結合の再同期化

BaseView 表の結合と、対応するデータベース表の結合を同期化するには、次の手順を実行します。

1. 同期化する結合で接続されている BaseView の表を選択します。複数の表を選択するには、[Ctrl] キーを使用します。
2. 「Resynchronize Table」ボタンをクリックします。
「Resynchronize BaseView Tables」ダイアログ・ボックスが表示されます。このダイアログ・ボックスでは、データベースへの接続に必要な情報は編集できません。
3. 別のタイプのデータベース・ディクショナリで記述された表を含めるには、ドロップダウン・リストから該当するデータ・ディクショナリ・タイプを選択します。使用している環境のログイン情報を入力して、「OK」をクリックします。
4. BaseView で選択されているデータベースの表のリストを表示するには、「Tables」ボタンをクリックします。
表のリストには、表が BaseView またはデータベース（あるいはその両方）に存在するかどうか、アイコンによって示されます。表のアイコンは BaseView を表し、円柱のアイコンはデータベースを表します。
5. ユーザー表以外の表の結合を再同期化するには、「Table Types」セクションで該当するチェック・ボックスを選択します。
6. データベース表から BaseView 表に結合を追加するには、「Add new joins to the BaseView」を選択します。
7. データベースに存在しない結合を BaseView から削除する場合は、「Remove obsolete/unmatched joins from the BaseView」を選択します。
8. 「OK」をクリックします。
BaseView エディタによって、選択された BaseView 表の結合とデータベースが同期化され、ダイアログ・ボックスに結果が表示されます。
9. 「OK」をクリックして変更を BaseView に適用するか、または「Cancel」をクリックして変更を破棄します。

MetaView の設計

ここで説明する内容は次のとおりです。

- [MetaView](#) での作業
- [MetaView](#) の編集

MetaView での作業

MetaView は、通常、1つの BaseView に基づき、BaseView のサブセットについて簡略化されたビューを提供します。1つの BaseView から複数の MetaView を作成することで、異なるユーザー・グループまたは異なる状況に応じた様々なビューを提供できます。たとえば、マーケティング・ビュー、経理ビューおよび販売ビューは、同じ BaseView の異なる MetaView にすることができます。

MetaView には、複数の BaseView からの Part を含めることができます。複数の BaseView からの Part をプランで結合するには、データ・フローに Join Transform を追加します。データ・フローの詳細は、「[データ・フロー・エディタの使用](#)」を参照してください。データ・フロー・エディタへのアクセスは、ロール権限によって制限されます。ロール権限およびセキュリティ・グループは、Oracle BAM Enterprise Link Admin を使用して Oracle BAM Enterprise Link 管理者によって割り当てられます。

MetaView は、Parts Bin に表示されます。初期状態では、表は Category として、フィールドは Part として表示されます。Category および Part の名前の変更、アイテムの再配置およびアイテムの削除を実行できます。アイテムを削除すると、そのアイテムは MetaView のみから削除されます。また、新しい Category を作成し、BaseView からフィールドをドラッグして Part を追加できます。MetaView では、計算 Part の作成と、Part に対する制約の追加を実行できます。

ここで説明する内容は次のとおりです。

- [MetaView の作成](#)
- [MetaView の保存](#)
- [MetaView の選択](#)
- [MetaView の削除](#)
- [MetaView のコピー](#)

MetaView の作成

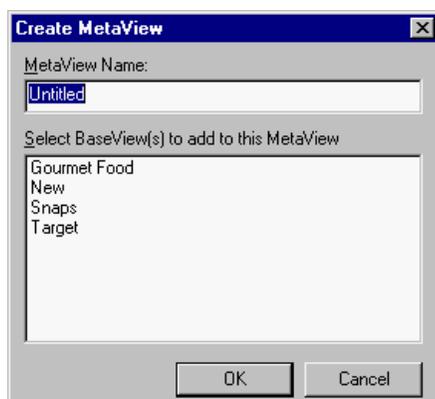
MetaView を作成する場合、BaseView 内のすべてのアイテムを含めて始めるか、または空の MetaView から始めることができます。BaseView の表と列をすべて追加しない場合は、空の MetaView から始めます。MetaView を作成するには、MetaView を作成するための権限を持つロールが割り当てられている必要があります。

デフォルトの **Category** と **Part** で新しい **MetaView** を作成するには、次の手順を実行します。

1. 「**Parts Bin**」 ボタンをクリックして、**Parts Bin** を開きます。
2. 「**Search**」 ボタンをクリックして、「**Create New MetaView**」 を選択します。



「**Create MetaView**」 ダイアログ・ボックスが表示されます。



3. **MetaView** の名前を入力します。
4. この **MetaView** に追加する **BaseView** を選択します。選択は複数でもかまいません。
5. 「**OK**」 をクリックします。

MetaView が作成されます。選択した **BaseView** からの表が **MetaView** の **Category** としてすべて追加され、**BaseView** 表の列が **MetaView** の **Part** としてすべて追加されます。**BaseView** のサイズと複雑さによっては、時間がかかることがあります。

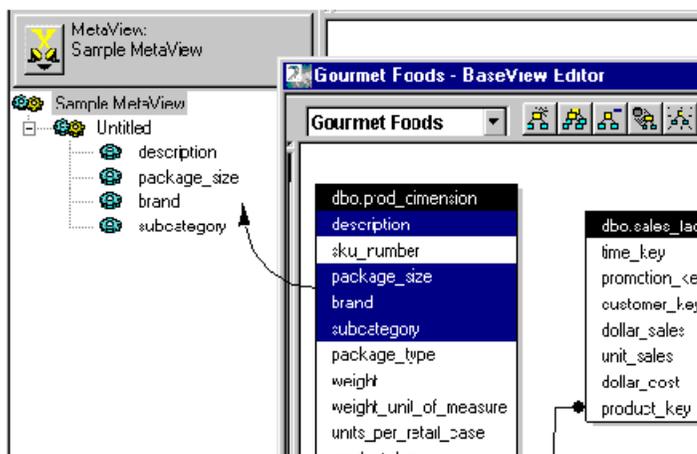
MetaView を作成した後で、さらに表を追加できます。また、異なる **BaseView** からも表を追加できます。

新しい空の **MetaView** を作成し、**Part** を追加するには、次の手順を実行します。

1. 「**Parts Bin**」 ボタンをクリックして、**Parts Bin** を開きます。
2. 「**Search**」 ボタンをクリックして、「**Create New MetaView**」 を選択します。
3. **MetaView** の名前を入力しますが、**BaseView** は選択しないでください。

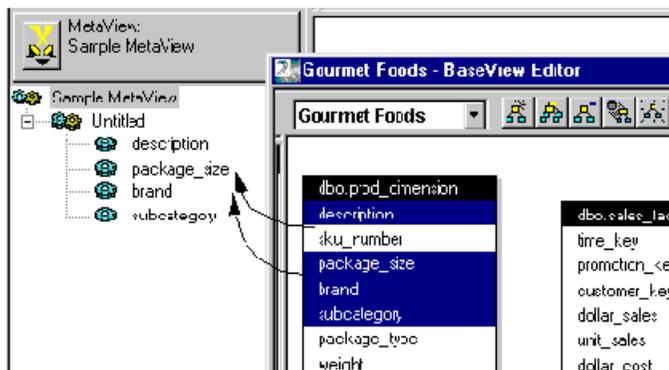
4. 「OK」 をクリックしてダイアログ・ボックスを閉じます。

空の MetaView が作成され、既存の BaseView から列を追加できるようになりました。



5. 「Tools」 → 「BaseView Editor」 を選択し、「BaseView」 ドロップダウン・リストから BaseView を選択します。
6. [Ctrl] キーを押しながら、BaseView エディタの表から列を Parts Bin にドラッグします。これらの列は、Untitled と呼ばれる Category で、MetaView の Part になります。

また、「BaseView Tables」 リストから Parts Bin に表をドラッグして、Category を MetaView に追加することもできます。



MetaView にアクセスするユーザーは、MetaView を読み取る権限のあるセキュリティ・グループに属している必要があります。

Category 名と Part 名を編集できます。詳細は、[67 ページの「Part および Category の名前の変更」](#) を参照してください。

MetaView の保存

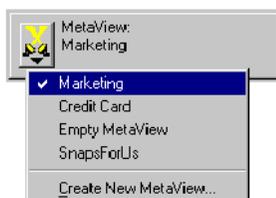
MetaView に行った変更は、自動的にすぐ保存されます。

MetaView の選択

MetaView を表示するには、MetaView に対する読取り権限があるセキュリティ・グループに属している必要があります。

MetaView を選択するには、次の手順を実行します。

1. 「Parts Bin」 ボタンをクリックして、Parts Bin を開きます。
「Search」 ボタンの横に、現行の MetaView の名前が表示されます。
2. 「Search」 ボタンをクリックして、MetaView を選択します。



Parts Bin に、MetaView が表示されます。多数の MetaView に対しアクセスを所有している場合は、「Search」 ボタン・メニューに、メニュー項目「More」が表示されます。「More」を選択して、アクセスを所有しているすべての MetaView のリストを表示します。

MetaView の削除

MetaView を削除するには、MetaView に対する書込み権限があるセキュリティ・グループに属している必要があります。MetaView を削除する前に、ユーザーがその MetaView にアクセスしていないことを確認します。

注意： MetaView を削除すると、その MetaView からの Part を使用しているプランが無効になります。

MetaView を削除するには、次の手順を実行します。

1. 「Search」 ボタンをクリックして、削除する MetaView を選択します。
2. MetaView 名を右クリックし、「Delete」を選択します。
メッセージによって、選択した MetaView の削除について確認が求められます。
3. 「OK」 をクリックして、MetaView を削除します。
MetaView は、[Delete] キーを押しても削除できません。

MetaView のコピー

MetaView に、名前を変更した Part と、制約と計算を使用して作成した Part が含まれている場合は、MetaView をコピーする場合があります。このような場合、これらのアイテムを再び作成する必要はありません。かわりに、Part を編集して、新しい MetaView 名を含めます。制約と計算を使用した Part の詳細は、71 ページの「制約の追加」および 73 ページの「計算 Part の作成」を参照してください。

MetaView をコピーするには、その MetaView に対するコピー権限があるセキュリティ・グループに属している必要があります、MetaView を作成する権限を持つロールが割り当てられている必要があります。

MetaView をコピーするには、次の手順を実行します。

1. 「**Search**」 ボタンをクリックして、コピーする MetaView を選択します。
2. Parts Bin で右クリックし、「**Copy MetaView**」 を選択します。

MetaView が新しい MetaView にコピーされ、その名前はオリジナルの名前の前に Copy of が付いたものになります。新しい MetaView の名前を変更するには、Parts Bin を右クリックして、「**Edit MetaView**」 を選択します。新しい名前を入力し、「**OK**」 をクリックします。

MetaView の編集

MetaView を作成した後で、その MetaView を編集できます。外観の変更、不要なアイテムの削除、および新しいアイテムの追加を実行できます。MetaView を編集するには、MetaView に対する読取りおよび書込み権限があるセキュリティ・グループに属している必要があります。

ここで説明する内容は次のとおりです。

- [MetaView の名前の変更](#)
- [MetaView への BaseView の追加](#)
- [BaseView からの列の追加](#)
- [MetaView 内の Category の展開](#)
- [MetaView からの Part と Category の削除](#)
- [Part および Category の名前の変更](#)
- [Part および Category の再配置](#)
- [Category の追加](#)
- [Part のプロパティの表示](#)
- [制約の追加](#)
- [計算 Part の作成](#)

MetaView の名前の変更

MetaView の名前を変更するには、次の手順を実行します。

1. 「Search」 ボタンをクリックして、編集する MetaView を選択します。
2. MetaView の名前をゆっくりと 2 回クリックします。
3. 新しい名前を入力し、[Enter] キーを押します。

また、MetaView の名前を右クリックして「Rename」を選択し、MetaView の名前を変更することもできます。

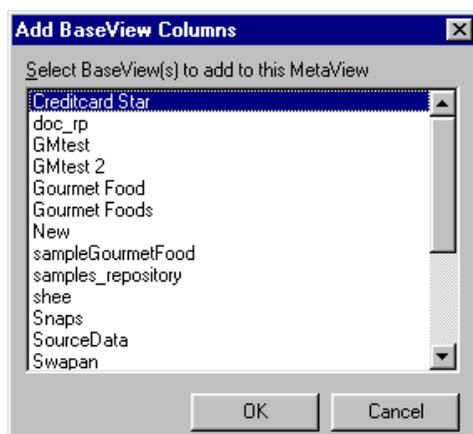
計算 Part またはオリジナルの MetaView の名前を含む制約がある Part を作成した場合は、Part 構文を編集して、MetaView の新しい名前を含めるようにする必要があります。制約と計算を使用した Part の詳細は、[71 ページの「制約の追加」](#) および [73 ページの「計算 Part の作成」](#) を参照してください。

MetaView への BaseView の追加

1 つの MetaView には、複数の BaseView からの表を含めることができます。Join Transform を使用すると、個別の BaseView からの表を同じプランに含めることができます。Transform の詳細は、[85 ページの「プランへの Transform の追加」](#) を参照してください。

BaseView を MetaView に追加するには、次の手順を実行します。

1. 「Search」 ボタンをクリックして、編集する MetaView を選択します。
2. Parts Bin で右クリックして「Add BaseView Columns」を選択し、「Add BaseView Columns」ダイアログ・ボックスを表示します。



3. この MetaView に追加する BaseView を選択します。選択は複数でもかまいません。選択した BaseView からのすべて表が Category として MetaView に追加されます。
4. 「OK」 をクリックしてダイアログ・ボックスを閉じます。

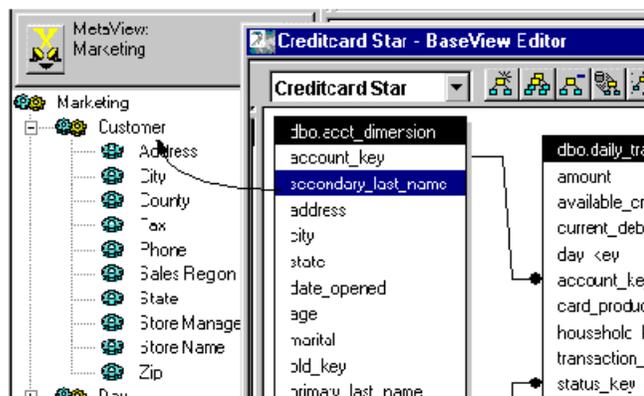
BaseView 表を追加した後で、使用しない Category を削除できます。これによって、BaseView が変更されることはありません。

BaseView からの列の追加

BaseView から Parts Bin に列をドラッグして、Part を MetaView に追加します。Part を追加する場合は、既存の Category に配置したり、新しい Category を作成できます。

BaseView から MetaView に列を追加するには、次の手順を実行します。

1. 「Tools」 → 「BaseView Editor」 を選択し、ドロップダウン・リストから BaseView を選択します。



2. BaseView から Parts Bin に列をドラッグします。
 - 列を余白の領域にドロップすると、新しい Part に対して Untitled という Category が作成されます。
 - 列を Category アイコンにドロップすると、その Category に追加されます。

列は、MetaView 内の Part になります。複数の列を MetaView に追加するには、[Ctrl] キーを押しながら BaseView 内の列をクリックします。その後で、それらの列を MetaView にドラッグします。

MetaView 内の Category の展開

Parts Bin で MetaView を最初に表示すると、Category は閉じた状態で表示されます。Part を表示するには、Category を開きます。

Category を開くには、次のいずれかを実行します。

- Category をダブルクリックします。
- Category 名の横にあるツリー・コントロールをクリックします。
- 右クリックして、「Expand All Categories」を選択します。

MetaView からの Part と Category の削除

MetaView から Part または Category を削除しても、BaseView には影響しません。Part を削除すると、削除されたその Part を含むプランがすべて無効になります。Category を削除すると、その Category 内のすべての Part が削除されます。

MetaView から Category または Part を削除するには、次の手順を実行します。

1. Parts Bin でアイテムをクリックして [Delete] キーを押すか、Parts Bin で右クリックして「Delete」を選択します。
2. メッセージによって、アイテムの削除について確認が求められます。「OK」をクリックします。

確認メッセージは、カスタマイズ可能なユーザー・オプションです。詳細は、[23 ページの「Bin の設定の変更」](#)を参照してください。

Part および Category の名前の変更

Parts Bin 内のアイテムの名前を変更するには、「Create MetaViews」ロール権限が必要です。

Parts Bin 内のアイテムの名前を変更するには、次の手順を実行します。

1. 名前をゆっくりと 2 回クリックします。
2. 新しい名前を入力し、[Enter] キーを押します。

また、アイテムを右クリックし、「Rename」を選択して、Parts Bin 内のアイテムの名前を変更することもできます。

注意： MetaView には同じ表またはフィールドのコピーを複数含めることができますが、Category には MetaView 内で一意の名前を、Part には Category 内で一意の名前を指定する必要があります。重複がある場合に、これらの Part を含むプランを実行すると、結果のデータに誤りが発生する可能性があります。

Part および Category の再配置

MetaView 内の様々な Category に Part を移動し、アイテムを論理的にグループ化できます。Category は、アルファベット順にソートされます。

Part を Category にコピーするには、次の手順を実行します。

➤ [Ctrl] キーを押しながら Part を新しい Category にドラッグします。

Part を Category に移動するには、次の手順を実行します。

➤ Part を新しい Category にドラッグします。

Category の追加

新しい Category を MetaView に追加するには、次の手順を実行します。

1. Parts Bin で右クリックして、「**New Category**」を選択します。

Untitled という新しい Category が、Parts Bin の他の Category の下に作成されます。

2. 名前をゆっくりと 2 回クリックします。

3. 新しい名前を入力し、[Enter] キーを押します。

次のいずれかの操作を実行すると、Part を Category に追加できます。

- 他の Category から Part を移動する
- BaseView エディタから Part をコピーする
- 計算 Part を作成する

計算 Part の詳細は、[73 ページの「計算 Part の作成」](#)を参照してください。

Part のプロパティの表示

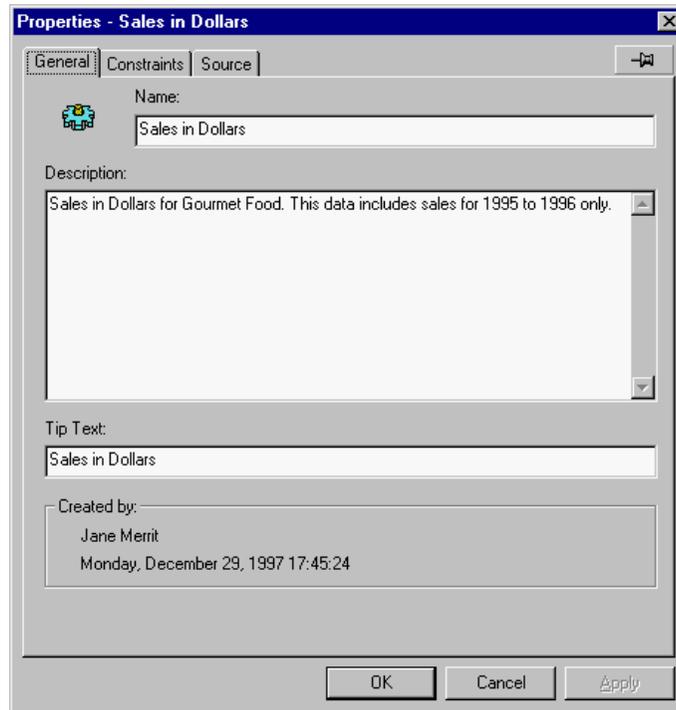
「Part Properties」ダイアログ・ボックスには、選択した Part の詳細が表示されます。また、Part のプロパティには、説明およびヒントのテキストを追加することもできます。

Part のプロパティを表示するには、次の手順を実行します。

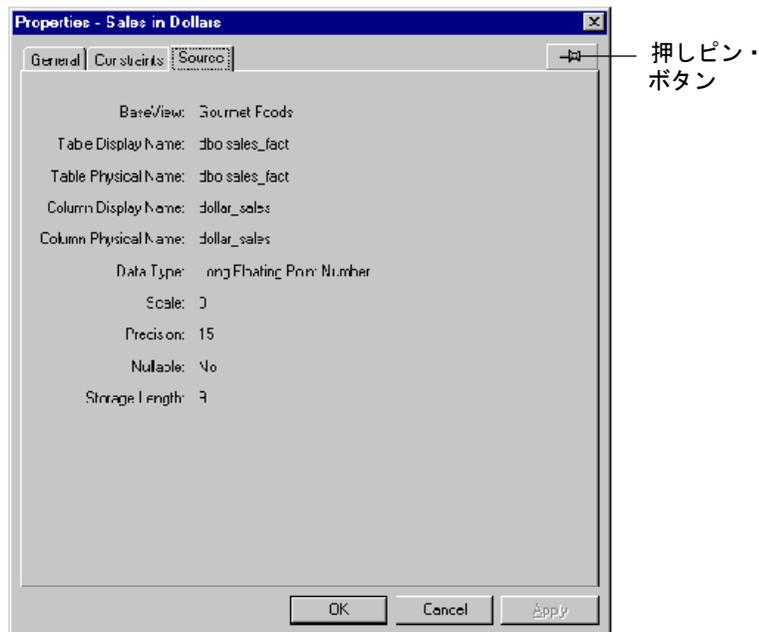
1. 「**Parts Bin**」 ボタンをクリックして、Parts Bin を開きます。

2. Parts Bin 内の Part を右クリックして、「**Properties**」を選択します。

「Properties」ダイアログ・ボックスが表示されます。「General」タブには、Part 名、説明およびヒントのテキスト用のフィールドがあります。



3. 「Source」タブをクリックして、さらに Part のプロパティを表示します。



「Part Properties」ダイアログ・ボックスを開いたままにして、他の Part を選択し、そのプロパティを表示できます。また、押しピン・ボタンをクリックすると、他の Part を選択した場合にも、元の Part のダイアログ・ボックスにフォーカスを固定したままにできます。

次の表に、プロパティの説明を示します。

プロパティ	説明
Name	Part に表示される名前。
Description	Part の説明。
Tip Text	Part にカーソルを置いたときに Part で表示される説明。
Created by	Part を作成したユーザーと日時。
Constraints	Part に適用されている制約。
BaseView	Part の追加元の BaseView。
Table Display Name	その Part の BaseView におけるソース表の表示名。
Table Physical Name	その Part の BaseView におけるソース表。
Column Display Name	その Part の BaseView におけるソース列の表示名。
Column Physical Name	その Part の BaseView におけるソース列。
Data Type	列のデータベース・タイプ。計算 Part のこの値は、「Calculation」タブで編集できます。
Scale	DECIMAL 列における列のスケール。
Precision	DECIMAL 列における列の精度。
Nullable	列で NULL 値を許可するかどうかを決定する値（「Yes」または「No」）。
Storage Length	列の格納長。

制約の追加

「Properties」ダイアログ・ボックスに制約を追加すると、SQL Query の WHERE 句に制約が追加されます。MetaView のグローバル制約を入力するための構文は、Part に対する制約の構文と同じです。

ここで説明する内容は次のとおりです。

- [制約を定義するための構文](#)
- [MetaView への制約の追加](#)
- [Part への制約の追加](#)

制約を定義するための構文

MetaView および Part に対する制約の定義では、同じ構文を使用します。制約に含めるフィールドは、Part またはデータベース列として指定できます。制約を入力するための構文は、次のとおりです。

- MetaView 内の Part を指定する場合：
[[Part:"MetaView"."Category"."PartName"]]
- データベース表内の列を指定する場合：
[[Column:"BaseView"."Table"."ColumnName"]]

二重カッコは必須です。名前に空白またはピリオドが含まれる場合、引用符は必須です。式に SQL 論理演算子を使用して、制約に追加の条件を含めることができます。次に例を示します。

```
[[Part:Accounting.Customer.State]] = 'TN' OR  
[[Part:Accounting.Customer.State]] = 'CA'
```

また、カスタム・プラン・プロパティを使用する制約を定義することもできます。プラン・プロパティの詳細は、[115 ページの「計算 Part のカスタム・プラン・プロパティの使用」](#)を参照してください。

MetaView への制約の追加

「Constraints」タブを使用すると、MetaView のグローバル制約を入力できます。グローバル制約は、MetaView からの Part がプランで使用される場合に SQL 文で使用されます。

制約を MetaView に追加するには、次の手順を実行します。

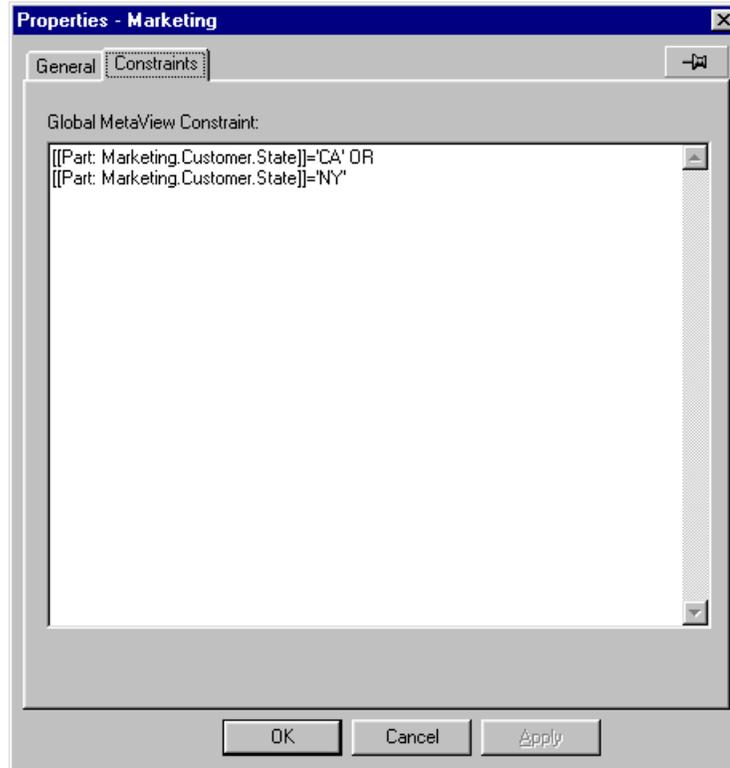
1. MetaView を右クリックし、「Properties」を選択します。
2. 「Constraint」タブをクリックします。
3. 押しピン・ボタンをクリックして、その MetaView のダイアログ・ボックスにフォーカスを固定します。

4. Parts Bin から「Constraint」フィールドに Part をドラッグします。

たとえば、Store Name Part を使用したプランの結果を、California および New York の店舗に制限するには、State をフィールドにドラッグし、次のようにテキストを追加します。

[[Part:Marketing.Customer.State]] = 'CA' OR

[[Part:Marketing.Customer.State]] = 'NY'



5. 「OK」をクリックして、ダイアログ・ボックスを閉じます。

Part への制約の追加

制約を MetaView 内の Part に追加するには、次の手順を実行します。

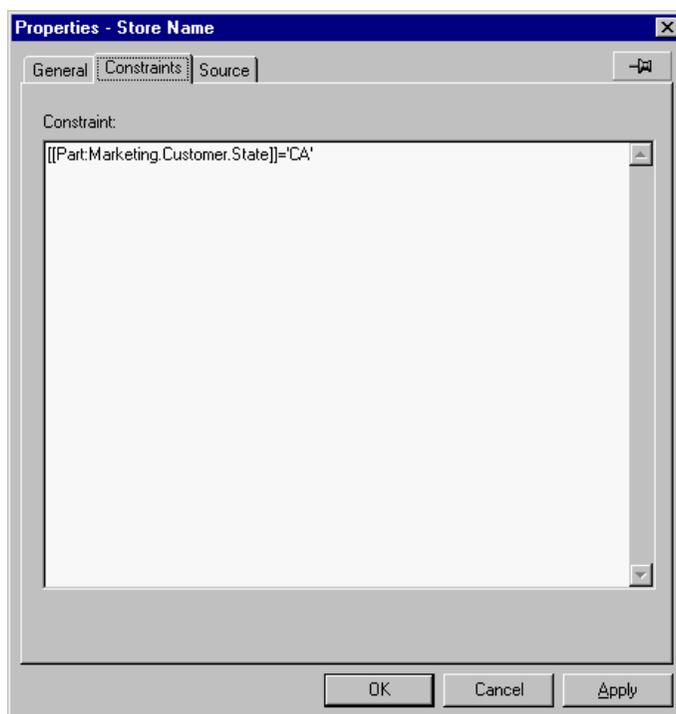
1. Part を右クリックし、「Properties」を選択します。
2. 「Constraint」タブをクリックします。
3. 押しピン・ボタンをクリックして、元の Part のダイアログ・ボックスにフォーカスを固定します。

4. Parts Bin から「Constraint」フィールドに Part をドラッグします。

たとえば、Store Name Part を使用したプランの結果を、California の店舗に制限するには、State をフィールドにドラッグし、次のようにテキストを追加します。

[[Part:Marketing.Customer.State]] = 'CA'

また、Part をドラッグするかわりに、制約の式をすべて入力することもできます。



5. 「OK」をクリックして、ダイアログ・ボックスを閉じます。

計算 Part の作成

入力フィールド、演算子および関数を組み合わせて新しい値を生成するように、計算 Part を MetaView に追加できます。

ここで説明する内容は次のとおりです。

- [計算 Part を定義するための構文](#)
- [計算 Part の作成](#)
- [計算 Part の編集](#)

計算 Part を定義するための構文

計算に含めるフィールドは、Part またはデータベース列として指定できます。計算 Part を入力するための構文は、次のとおりです。

- MetaView 内の Part を指定する場合：
[[Part:"MetaView"."Category"."PartName"]]
- データベース表内の列を指定する場合：
[[Column:"BaseView"."Table"."ColumnName"]]

二重カッコは必須です。名前に空白またはピリオドが含まれる場合、引用符は必須です。

また、カスタム・プラン・プロパティを使用する計算 Part を定義することもできます。プラン・プロパティの詳細は、[115 ページの「計算 Part のカスタム・プラン・プロパティの使用」](#)を参照してください。

フィールド構文を入力する場合、演算子はその両側に空白を入れて区切ります。計算 Part では、次の演算子を使用できます。

演算	記号
加算	+
減算	-
乗算	*
除算	/
開きカッコ	(
閉じカッコ)

計算 Part では、次の関数を使用できます。

関数	構文
合計	SUM
平均	AVG
最小値	MIN
最大値	MAX
カウント	COUNT
区別したカウント	COUNT DISTINCT

SQL Query に計算 Part を使用する場合、SQL を生成するためには、基礎となる表に基づく 1 つ以上の Part を含める必要があります。これ以外の場合、問合せはユーザー入力による SQL と同様に扱われます。

計算 Part の作成

MetaView に計算 Part を作成するには、次の手順を実行します。

1. Parts Bin で、計算 Part を追加する Category をクリックします。
2. 右クリックして「**New Calculated Field**」を選択します。
新しい Part が Parts Bin に追加され、その Part の「**Properties**」ダイアログ・ボックスが表示されます。
3. 「**General**」タブで、計算 Part の新しい名前、説明およびヒントのテキストを入力します。
4. 「**Calculation**」タブをクリックします。
5. 計算によって作成されるフィールドのタイプについて、ドロップダウン・リストからデータ型を選択します。

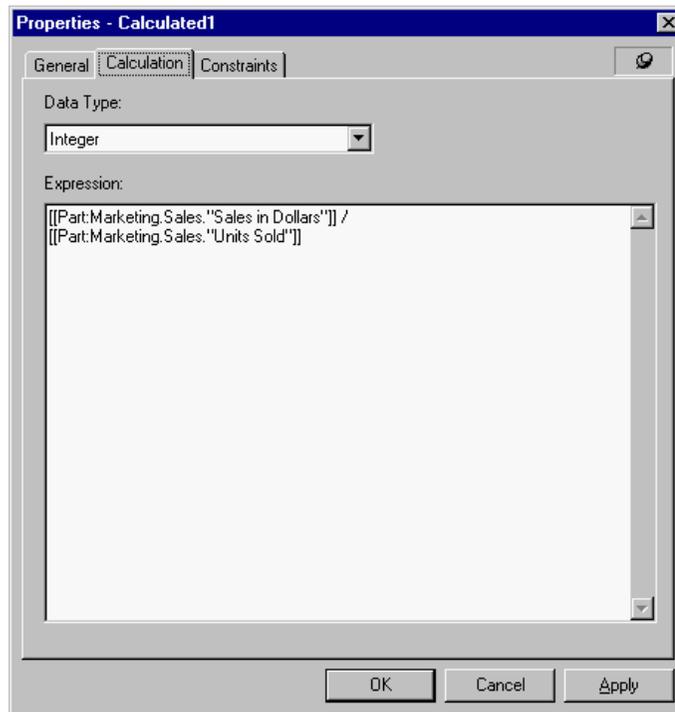
データ型	説明
Date and Time	式の結果が日時データ型の場合は、「Date and Time」を選択します。
Integer	式の結果が 64,000 未満の正の整数値の場合は、「Integer」を選択します。
Long Floating Point Number	式の結果が $\pm 2,000,000,000$ までの正または負の実数（少数を含む）の場合は、「Long Floating Point Number」を選択します。このデータ型は、C の double データ型に一致します。
Long Integer	式の結果が $\pm 2,000,000,000$ までの正または負の整数の場合は、「Long Integer」を選択します。このデータ型は、C の 32-bit integer データ型に一致します。
String	式の結果が英数字フィールドの場合は、「String」を選択します。

6. 押しピン・ボタンをクリックしてから、Part をダイアログ・ボックスにドラッグします。

7. Parts Bin から「Expression」フィールドに Part をドラッグして、計算を作成します。演算子を入力する必要があります。次に例を示します。

[[Part:Marketing.Sales."Sales in Dollars"]] / [[Part:Marketing.Sales."Units Sold"]]

また、Part をドラッグするかわりに、計算 Part の式を入力することもできます。



8. 必要に応じて、「Constraints」タブをクリックし、制約を入力します。
この例の場合、次の制約を使用すると、0（ゼロ）による除算が回避されます。

[[Part:Marketing.Sales."Units Sold"]] <> 0

9. 「OK」をクリックしてダイアログ・ボックスを閉じます。

計算 Part の編集

計算 Part を編集するには、次の手順を実行します。

1. 計算 Part を右クリックして「Properties」を選択し、「Properties」ダイアログ・ボックスを表示します。
2. 「Calculation」タブをクリックし、計算 Part の式を編集します。
3. 「OK」をクリックしてダイアログ・ボックスを閉じます。

データ・フロー・エディタの使用

ここで説明する内容は次のとおりです。

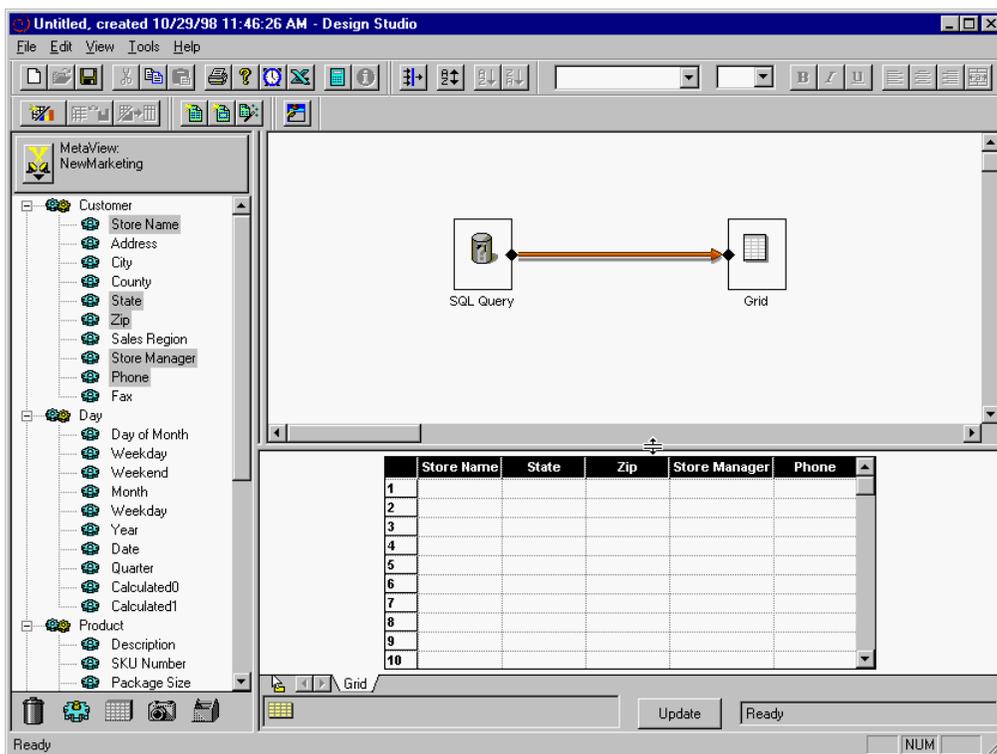
- データ・フロー・エディタでのプランの表示
- Transform の使用

データ・フロー・エディタでのプランの表示

データ・フロー・エディタには、プランのステップが表示されます。データ・フロー・エディタを使用して、プランの構造を表示し、プランの拡張機能にアクセスします。メイン・ウィンドウには、プランのステップが左から右への線の形式で表示されます。大規模なプランを移動する場合は、プランのステップをツリー構造で表示するパネルを開くこともできます。

データ・フロー・エディタを表示するには、次の手順を実行します。

- ワークスペースの上部のハンドル（カーソルが上下の矢印になる）にカーソルを置き、マウスをワークスペースに向かって下にドラッグします。



データ・フロー・エディタの表示は、Oracle BAM Enterprise Link Admin でのロール権限です。ロール権限の割当ての詳細は、『Oracle Business Activity Monitoring Enterprise Link 管理者ガイド』を参照してください。

ワークスペースに Part をドラッグしてプランを作成する場合、データ・フローには次の 2 つのステップがあります。

- データを選択するための SQL Query
- データを表示するための Grid



Tool Bin から Transform を使用して、このプランにステップを追加できます。Transform をデータ・フロー・エディタにドラッグして、入力と出力を接続します。Transform の詳細は、『Oracle Business Activity Monitoring Enterprise Link Transforms リファレンス・ガイド』を参照してください。

各 Transform では、特定のタイプの処理が提供されます。Transform に設定するパラメータがある場合は、Transform をダブルクリックして、ダイアログ・ボックスを表示します。このプランを保存する場合は、データ・フローと、設定したすべてのパラメータを保存します。

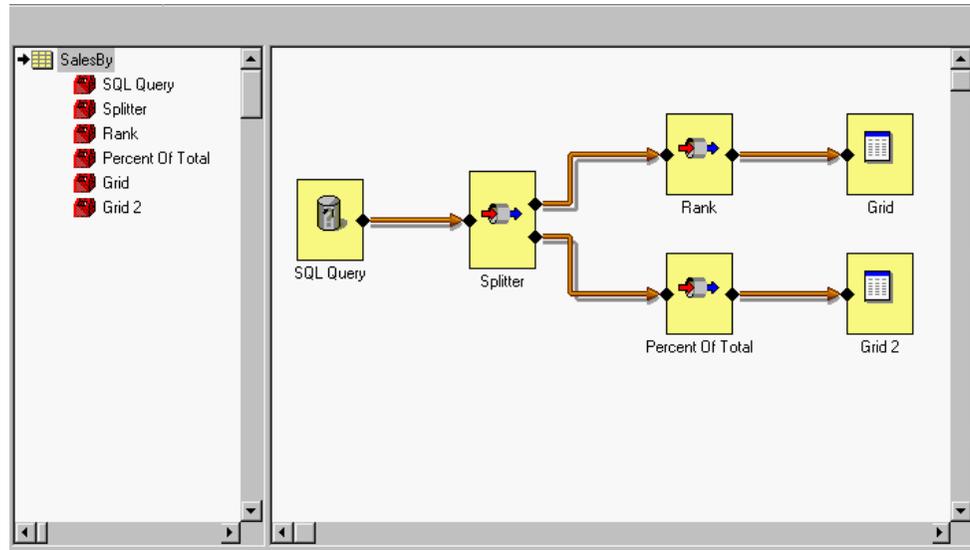
大規模なプランを移動する場合は、プランのステップをツリー構造で表示するパネルを開くことができます。これは、複雑なプランで特定のステップに移動する場合に役立ちます。

- データ・フロー・エディタでのズーム操作
- データ・フローの印刷
- 複数のデータ・フローの作成
- データ・フロー・エディタからのプランの名前の変更

データ・フロー・エディタでツリーを開くには、次の手順を実行します。

- データ・フロー・エディタの左側のハンドル（カーソルが左右の矢印になる）にカーソルを置き、マウスをデータ・フロー・エディタに向かって右にドラッグします。

プランのステップがツリー構造で表示されます。



プランのステップを表示するには、ツリーでステップをクリックします。

データ・フロー・エディタでのズーム操作

データ・フロー・エディタで、プランのステップの表示を拡大または縮小できます。表示サイズは、データ・フローの印刷物には影響しません。ズーム・オプションにアクセスする場合、ステップが選択されていないことを確認します。

表示を拡大するには、次の手順を実行します。

1. データ・フローで右クリックし、「**Zoom In**」を選択します。
カーソルが、プラス (+) 記号が付いた虫メガネになります。
2. データ・フローでクリックして、表示を拡大します。倍率を上げるには、複数回クリックします。
3. データ・フローの特定の領域をズーム・インし、そこにフォーカスを固定するには、クリックするかわりに、選択ボックスをクリックしてドラッグします。

表示が拡大され、選択された領域が新しいビューに含まれます。

表示を縮小するには、次の手順を実行します。

1. データ・フローで右クリックし、「**Zoom Out**」を選択します。
カーソルが、マイナス (-) 記号が付いた虫メガネになります。
2. データ・フローでクリックして、表示を縮小します。さらに縮小するには、複数回クリックします。
3. 特定の領域をズーム・アウトし、そこにフォーカスを固定するには、クリックするかわりに、選択ボックスをドラッグします。

データ・フロー・エディタ・ウィンドウに合うようにプランをサイズ変更するには、次の手順を実行します。

- データ・フローで右クリックし、「**Zoom to Fit**」を選択します。

表示がサイズ変更され、プラン内のすべてのステップが表示されます。

表示を元のサイズに戻すには、次の手順を実行します。

- データ・フローで右クリックし、「**Normal Size**」を選択します。

データ・フローの印刷

データ・フローのステップを示すプランのコピーを印刷できます。印刷物は、様々なプランの比較や、**Design Studio** 以外でプランを参照するために使用できます。データ・フローを印刷するために、それをプランとして保存しておく必要はありません。

データ・フローを印刷するには、次の手順を実行します。

1. データ・フロー・エディタで右クリックし、「**Print**」を選択します。
Windows の印刷ダイアログ・ボックスが表示されます。デフォルトのプリンタが選択されています。
2. 異なるプリンタを使用するには、ドロップダウン・リストから選択します。
3. 印刷ファイルを作成する場合は、「**ファイルへ出力**」をクリックします。
4. 印刷するページ範囲と部数を指定します。
5. 「**OK**」をクリックします。

複数のデータ・フローの作成

複数のデータ・フローを同じプランに、または結果が2つのグリッドになる単一のデータ・フローに作成できます。

複数のデータ・フローを作成するには、次の手順を実行します。

1. 各データ・フローの **Transform** をデータ・フロー・エディタにドラッグします。
2. ステップを2つの完成したフローに接続します。それぞれには、ソースおよびシンク・ステップがあります。

ワークスペースの下部にある2つの「Grid」タブ間で切り替えて、結果を表示できます。

データ・フロー・エディタからのプランの名前の変更

データ・フロー・エディタからプランの名前を変更するには、次の手順を実行します。

1. データ・フロー・エディタで右クリックし、「**Rename Plan**」を選択します。
2. プランの新しい名前を入力します。

名前がタイトル・バーで変更されます。

Transform の使用

Part をワークスペースにドラッグしてプランを作成すると、SQL Query Transform および Grid Transform がプランに自動的に追加されます。これらの Transform の設定を変更したり、他の Transform を追加して複合的なプランを作成できます。他の Transform の使用の詳細は、オンライン・ヘルプを参照してください。

ここで説明する内容は次のとおりです。

- [SQL Query Transform](#)
- [Grid Transform](#)
- [プランへの Transform の追加](#)
- [「Transform」メニューの使用](#)
- [ステップのプロパティの編集](#)
- [サブプランの使用](#)
- [プランでの反復の使用](#)
- [ステップの入力と出力の表示](#)
- [Transform の共有](#)

SQL Query Transform

SQL Query Transform は、データ・フローのソースとして使用されます。1つのプランに、複数のデータ・フローを含めることができます。Parts Bin からワークスペースにアイテムをドラッグしてプランを作成すると、SQL Query ソースおよび Grid シンクのあるデータ・フローが自動的に作成されます。

SQL 文の構成は、Parts Bin からワークスペースにアイテムをドラッグするか、または「SQL Editor」ダイアログ・ボックスに SQL SELECT 文を入力して行います。使用している RDBMS サーバー・ソフトウェアがサポートしている SELECT 文は、いずれも入力できます。

ここで説明する内容は次のとおりです。

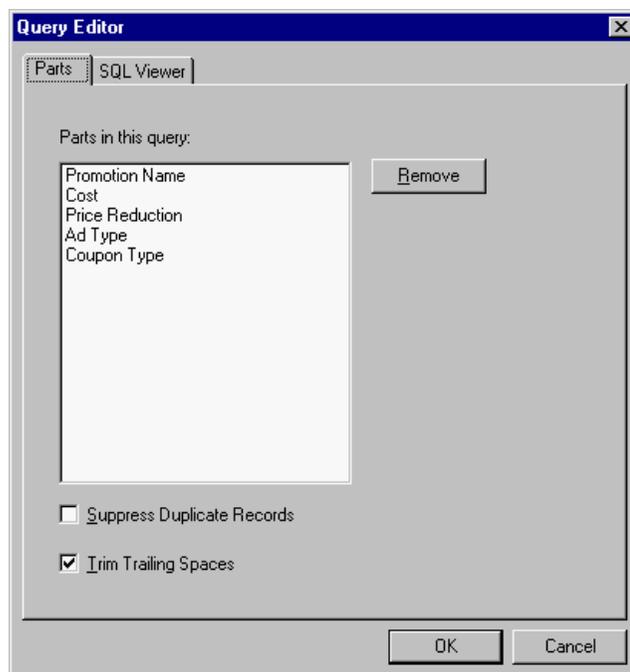
- [選択されている列の削除](#)
- [問合せの表示](#)
- [問合せの編集](#)

選択されている列の削除

クエリー・エディタの「Parts」タブを使用して、SQL を編集せずに問合せから列を削除します。

SQL Query ソースの SQL SELECT 文から列を削除するには、次の手順を実行します。

1. データ・フローの SQL Query ソースを右クリックし、「**Query Editor**」を選択します。
また、SQL Query ソース・アイコンをダブルクリックすることもできます。「Parts」タブに、問合せの列がその Part 名によって示されます。



2. データ・フローから削除する列を選択し、「**Remove**」をクリックします。
3. 重複する行をプランから自動的に削除する場合は、「**Suppress Duplicate Records**」チェック・ボックスを選択します。
4. プラン内の各列から、末尾の空白を削除しない場合は、「**Trim Trailing Spaces**」の選択を解除します。
5. 「**OK**」をクリックして変更を保存し、ダイアログ・ボックスを閉じます。

データ・フローおよび Grid の両方から列が削除されます。

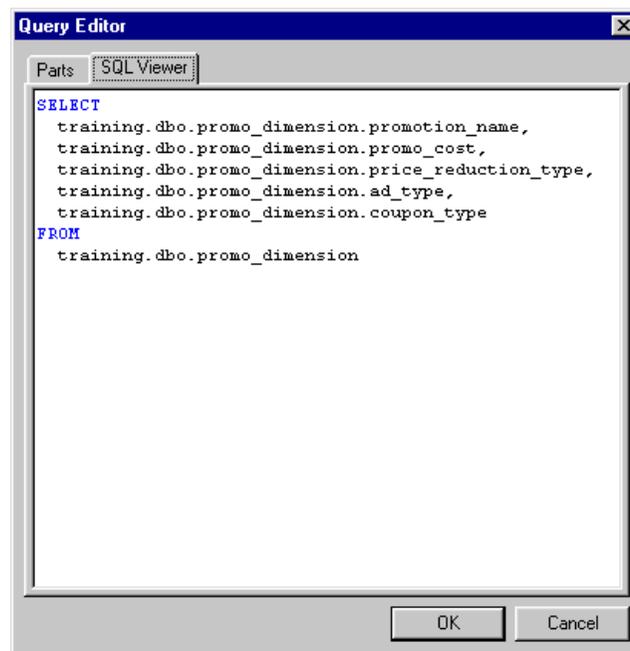
注意： Grid や他の表示 Transform から列を削除しても、SQL Query の SELECT 文からは削除されません。

問合せの表示

SQL Query Transform の SQL 文を表示するには、次の手順を実行します。

1. データ・フローの SQL Query ソースを右クリックして「**Query Editor**」を選択し、「**SQL Viewer**」タブをクリックします。

また、SQL Query ソース・アイコンをダブルクリックし、ダイアログ・ボックスを表示することもできます。



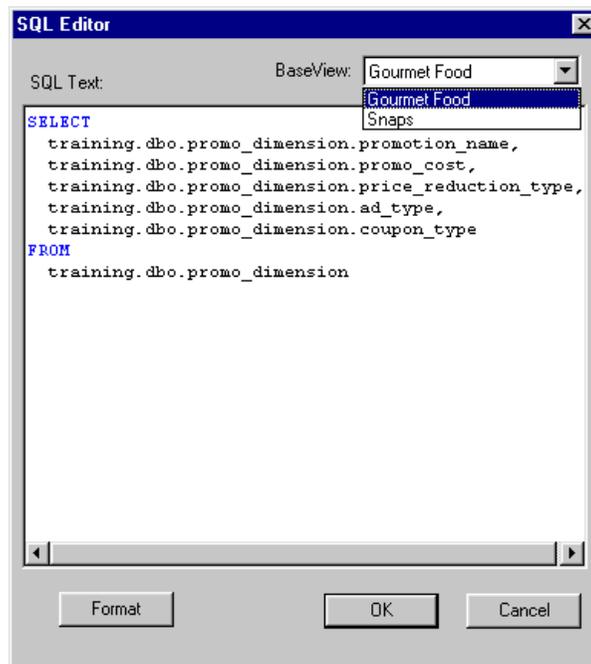
2. 「**OK**」をクリックして、「**Query Editor**」ダイアログ・ボックスを閉じます。

問合せの編集

SQL を入力または編集する場合は、SQL エディタを使用します。クエリー・エディタは、SQL 文の表示に使用します。

SQL 文を追加または編集するには、次の手順を実行します。

1. データ・フローの SQL Query ソースを右クリックして「SQL Editor」を選択し、「SQL Editor」ダイアログ・ボックスを表示します。



2. 問い合わせる BaseView を「BaseView」ドロップダウン・リストから選択します。SQL 文を入力または編集する場合は、BaseView を指定する必要があります。
3. SQL テキストを入力または編集します。アイテムを Parts Bin からワークスペースにドラッグしてプランを作成すると、このプランによって生成された SQL 文がダイアログ・ボックスに表示されます。
4. 「Format」をクリックして、SQL 文を構造化します。SQL の各キーワードが単一行に青字で配置され、SQL テキストが読みやすくなるように書式設定されます。
5. 「OK」をクリックして変更を保存します。

アイテムを Parts Bin からドラッグして問合せを作成した場合、これらの列を表示する Grid はワークスペースから消去されます。

注意： 編集した SQL は、Part をドラッグして作成した SQL 文と組み合わせることはできません。SQL 文を編集する場合は、Part をワークスペースにドラッグして、データ・フローに追加の SQL Query と Grid を作成します。

Grid Transform

Grid Transform は、データ・フローからの結果セットを示すためのシンクです。Grid では、形式を設定できるコラムの表示で、結果セットが表示されません。また、Grid は、Parts Bin から Part をドラッグして作成したプランのデフォルトのシンクです。

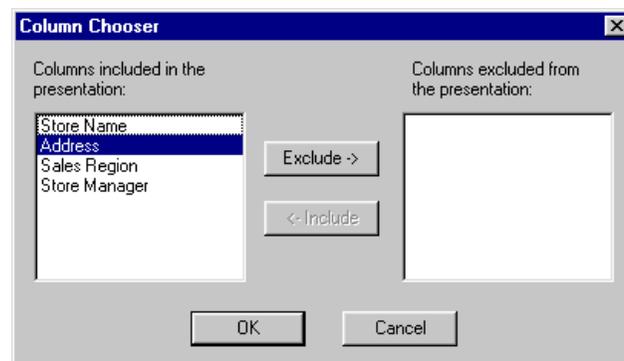
プランには、各データ・フローに独自の Grid または他のタイプのシンクを含めて、複数のデータ・フローを含めることができます。単一のデータ・フローに、複数のシンクを含めることもできます。複数のシンクがあるプランでは、各シンクが、ワークスペースの各シンク専用のタブに表示されます。

Grid から列を削除するには、列ヘッダーを選択して [Delete] キーを押すか、または「Column Chooser」を使用します。この方法で列を削除した場合、削除した列はプランのデータ・フローに含まれたままになります。

「Column Chooser」を使用して列を Grid から削除するには、次の手順を実行します。

1. Grid ステップを右クリックし、「Column Chooser」を選択します。

「Column Chooser」ダイアログ・ボックスが表示されます。



2. 左側のリストで列を選択して「Exclude」をクリックし、列を削除します。
3. 「OK」をクリックして変更を保存し、ダイアログ・ボックスを閉じます。

プランへの Transform の追加

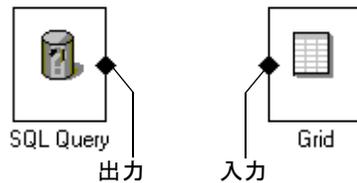
Transform のプランへの追加は、Transform をデータ・フロー・エディタにドラッグして行います。また、プランの作成は、Transform から開始することもできます。

Transform を単純なプランに追加するには、次の手順を実行します。

1. ハンドルをプルダウンして、データ・フロー・エディタを表示します。

ワークスペースにプランがすでにある場合、データ・フロー・エディタには SQL Query ステップと Grid ステップが表示されます。

2. Tool Bin からデータ・フロー・エディタに Transform をドラッグし、それをステップの間に配置します。



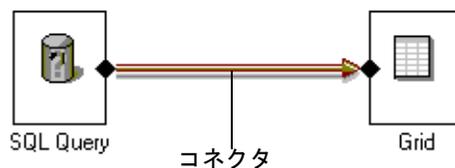
Design Studio によって、SQL Query ステップの出力が Transform ステップの入力に、Transform ステップの出力が Grid ステップの入力に接続されます。

3. Transform にダイアログ・ボックスがある場合は、Transform をダブルクリックして表示します。

ステップの接続

データ・フロー内のステップは、その接続点で接続する必要があります。Transform アイコンの側面にある黒色のひし形が接続点です。

通常、プランはソースで開始して、シンクで終了します。ステップが接続されている場合、データ・フローには、2つのステップ間のコネクタが表示されます。



Design Studio によって、データ・フロー内のほとんどのステップが自動的に接続されます。既存のデータ・フローに Transform をドラッグし、それをコネクタ付近にドロップすると、Transform が周辺のステップに接続されます。既存のデータ・フローでソース・ステップまたはシンク・ステップを置き換えると、新しいソースまたはシンクが隣接するステップに接続されます。また、ステップを削除すると、接続されているコネクタも削除されます。

データ・フローでステップを手動で接続する必要がある場合があります。

ステップを接続するには、次の手順を実行します。

- Transform の接続点をクリックして、隣りの Transform の接続点まで左から右にドラッグします。

接続の削除

プランから接続を削除するには、次の手順を実行します。

- コネクタをクリックして青色にし、[Delete] キーを押します。

接続が削除されます。

「Transform」メニューの使用

データ・フローの Transform についての情報は、表示または編集できます。プラン内のステップを右クリックして、「Transform」メニューを表示します。メニュー・コマンドは Transform によって若干異なりますが、このメニューには最も一般的なオプションが表示されます。表示される頻度の高いオプションについて、その情報を次の項に示します。設定オプションおよびプロパティ・オプションについては、106 ページの「Transform プロパティの使用」を参照してください。

注意： ダイアログ・ボックスがない Transform の場合、メニューから「Settings」、「Properties」および「Create Transform」オプションが除外されます。

ここで説明する内容は次のとおりです。

- ステップの名前の変更
- ステップの削除
- カスタム Transform の保存
- 「About Step」ダイアログ・ボックスの表示
- Transform の切り取り、コピーまたは貼付け

ステップの名前の変更

データ・フロー内のステップの名前は変更できます。シンクの名前を変更すると、ワークスペースの底部にあるタブの名前も新しい名前で表示されます。

データ・フロー・プランのステップの名前を変更するには、次の手順を実行します。

1. ステップを右クリックして「Rename Step」を選択し、ダイアログ・ボックスを表示します。
2. ステップの新しい名前を入力し、「OK」をクリックします。

ステップの削除

プランからステップを削除するには、次の手順を実行します。

- データ・フローで、Transform のアイコンをクリックして [Delete] キーを押します。また、ステップを右クリックして「Delete Step」を選択することもできます。

ステップが削除されます。

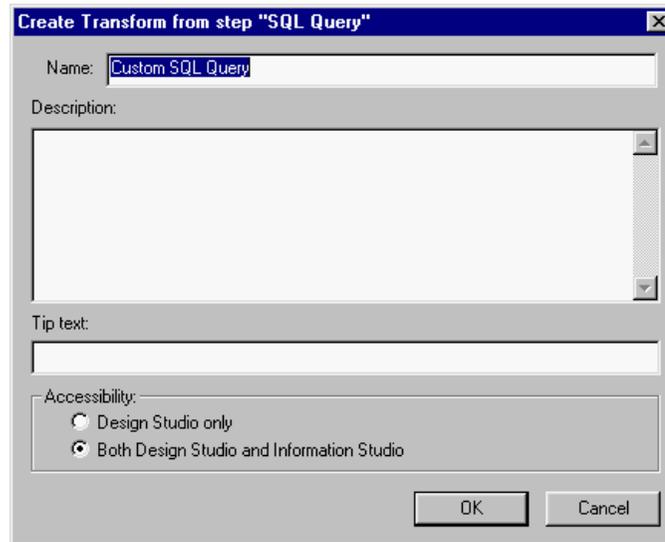
カスタム Transform の保存

ほとんどの Transform は、Tool Bin 内にカスタム Transform として保存できます。将来の使用に備えて設定を格納するために、Transform と設定を保存できます。たとえば、SQL 構文を編集した場合は、SQL Query Transform をカスタム Transform として保存できます。Tool Bin にアクセスできる他のユーザーも、この保存された Transform を使用できます。

カスタム Transform を保存するには、次の手順を実行します。

1. データ・フローで、Transform ステップを右クリックして「**Create Transform**」を選択します。

「Create Transform from Step "Name"」ダイアログ・ボックスが表示されます。



2. 新しい Transform の名前を入力します。
3. Transform の一般プロパティに表示される説明を入力します。
4. Tool Bin にツールのヒントとして表示されるテキストを入力します。
5. 新しい Transform を Design Studio のみに含めるか、または Design Studio と Information Studio の両方に含めるかに応じて、ラジオ・ボタンを選択します。他のユーザーも、Tool Bin から新しい Transform にアクセスできます。
6. 「OK」をクリックします。

Transform は、Tool Bin に表示されます。

カスタム Transform を作成すると、「Transform」ダイアログ・ボックスの現行の設定も保存されます。作成したカスタム Transform を削除するには、Tool Bin で選択して [Delete] キーを押します。

次の Transform はカスタム Transform として保存できません。Analysis、Columnar Report、Crystal Reports、Excel、Form Report、Grid、Save to Table、Splitter、StatView または Union。

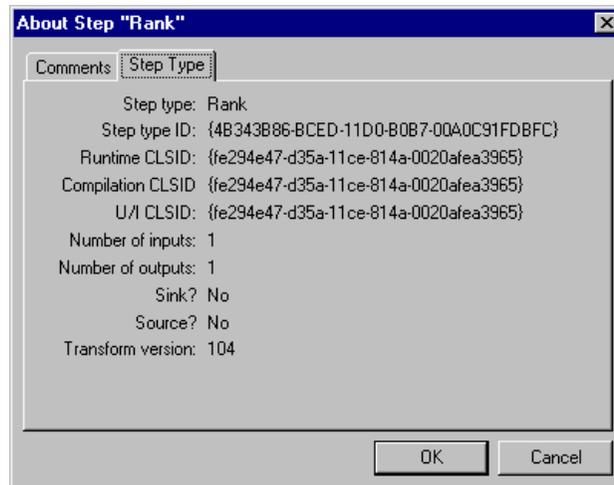
カスタム Transform の保存と、Transform SDK を使用したカスタム Transform の作成は異なります。カスタム Transform のプログラミングについては、『Transform Programmer's Guide』を参照してください。

「About Step」ダイアログ・ボックスの表示

「About Step」ダイアログ・ボックスには、テキストのコメントと、Transform についての内部情報が表示されます。この情報は、Transform の開発者にとって有益です。

Transform についての情報を表示するには、次の手順を実行します。

1. Transform を右クリックし、「About Step」を選択します。
2. 「Step Type」タブをクリックします。



「Comments」タブには、オプションの説明がコメントとして表示されます。コメントを追加または編集するには、ダイアログ・ボックスで入力を行い、「OK」をクリックします。

Transform の切り取り、コピーまたは貼付け

データ・フローの Transform は、切り取り、コピーおよび貼付けできます。Transform 定義をクリップボードに格納すると、すべての設定が保存されます。

データ・フローから Transform を削除し、それをクリップボードに格納するには、次の手順を実行します。

- データ・フローの Transform を右クリックし、「Cut」を選択します。

Transform 定義をコピーし、それをクリップボードに格納するには、次の手順を実行します。

- データ・フローの Transform を右クリックし、「Copy」を選択します。

クリップボードからデータ・フローに Transform 定義をコピーするには、次の手順を実行します。

- データ・フローの Transform を右クリックし、「Paste」を選択します。

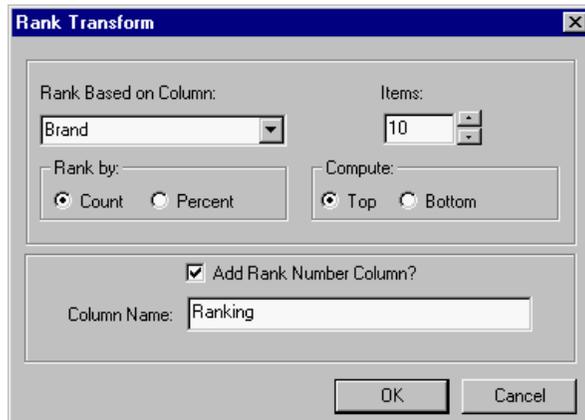
ステップのプロパティの編集

「Properties」ダイアログ・ボックスを使用して、Transform パラメータ値を編集できます。Conditional Splitter などの一部の Transform には、「Properties」ダイアログ・ボックスを表示するために、入力接続点に接続されるデータ・フローが必要です。他の Transform の場合、接続されていないと、変更済のダイアログ・ボックスが表示されます。たとえば、ダイアログ・ボックスで入力列を選択するためのドロップダウン・リストは、ユーザーが値を入力するテキスト・フィールドに置き換えられます。

ステップのプロパティを編集するには、次の手順を実行します。

1. データ・フローのステップを右クリックし、「Properties」を選択します。

「Properties」ダイアログ・ボックスが表示されます。また、ステップをダブルクリックしても、ステップのプロパティを表示できます。各ダイアログ・ボックスのパラメータは、Transform が実行する操作によって異なります。ステップのプロパティのダイアログ・ボックスの例として、Rank のダイアログ・ボックスを示します。



2. ステップのプロパティを編集します。

各 Transform については、『Oracle Business Activity Monitoring Enterprise Link Transforms リファレンス・ガイド』を参照してください。ステップを伴うカスタム・プランのプロパティの使用については、第6章「プロパティでの作業」を参照してください。

サブプランの使用

サブプラン Transform は、階層化されたプランをデータ・フロー・エディタで作成する場合に使用します。サブプラン Transform では、ステップをまとめてグループ化することによって、データ・フロー・エディタに一度に表示される記号の数が削減されます。サブプラン処理の一部として、データ・フローをサブプラン上の反復接続点に接続できます。反復接続点は、データに対して反復処理を実行する場合に使用します。通常では複数のステップおよびデータ・パスを必要とする、データ抽出タスクおよびデータ変換タスクの多くは、サブプラン Transform および反復処理を使用すると、より少ないステップで実行できます。

反復接続点を使用すると、各レコードまたはレコード・グループに対して繰り返す操作を定義できます。最初のレコードまたはレコードのグループに対するサブプランの処理が完了した後、サブプランに含まれているステップによって次のレコードまたはレコードのグループに対する処理が再開されます。

サブプランは、プランに無制限に追加できます。Transform は、そのタイプに関係なくサブプランに追加できます。表示シンクを反復サブプランに含めると、データが表示されるのは最初の反復からのみになります。反復を有効にしていないサブプランでは、表示シンクを制限なく使用できます。非表示シンクは、反復サブプランまたは非反復サブプランのいずれでも使用できます。

ここで説明する内容は次のとおりです。

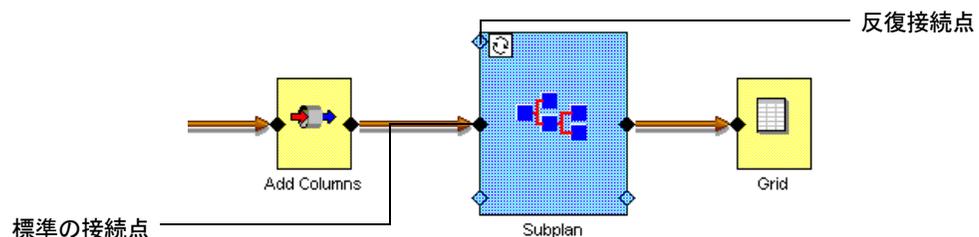
- データ・フローへのサブプランの追加
- サブプランの移動
- データ・フローからのサブプランの削除
- サブプランにおける反復プロパティの設定

データ・フローへのサブプランの追加

サブプランを追加するには、次の手順を実行します。

1. Transform Bin でサブプランをクリックし、その記号をデータ・フローにドラッグします。

Design Studio によって、サブプランの記号が追加され、その記号がデータ・フローに接続されます。Design Studio では、デフォルトで標準の入力接続点を使用されます。



2. データ・フローの入力を反復にするには、コネクタ線を削除し、新しいコネクタ線を反復接続点に接続します。

サブプランの反復の有効化については、[95 ページの「プランでの反復の使用」](#) および [96 ページの「サブプランにおける反復プロパティの設定」](#) を参照してください。

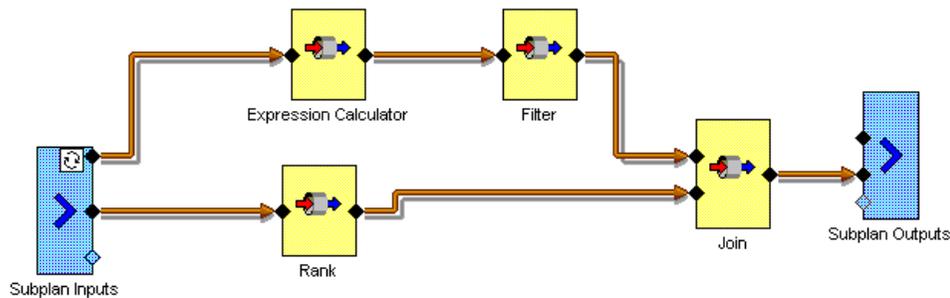
3. 他のデータ・フローをサブプランに接続する必要がある場合は、そのデータ・フローをサブプランの入力接続点に接続します。

データ・フローは、入力としてサブプランに無制限に接続できます。

4. サブプランの記号をダブルクリックします。

サブプランが開きます。サブプランのデータ・フローには、サブプラン入力記号とサブプラン出力記号のみが含まれます。

5. Transform を Transform Bin からドラッグして、サブプランに追加します。



6. 必要に応じて、コネクタ線をサブプラン入力の接続点に追加し、コネクタ線をデータ・フローの Transform と、サブプラン出力の接続点まで経路指定します。
7. 必要に応じて、他のサブプランの内部にサブプランを追加して、複数のプラン・レベルを形成できます。
8. サブプラン入力記号またはサブプラン出力記号をダブルクリックして、プランの最上位レベルに移動します。また、サブプランで右クリックし、「**Show Containing Plan**」を選択することもできます。
9. 最上位レベルのプランにおける接続線を、サブプランへと続く Transforms とシンクに接続します。

サブプラン出力の各接続点のデータ・フローがサブプラン内部のデータフローに接続されている場合、このサブプラン出力の各接続点には、上位レベルのサブプランに接続される対応するデータ・フローが必要です。詳細は、[93 ページの「サブプランでの作業」](#)を参照してください。

10. 「**Update**」をクリックしてプランを実行し、その出力を確認します。

新しいサブプランへのステップのグループ化

ステップのグループは、新しいサブプランに移動できます。

ステップを新しいサブプランにグループ化するには、次の手順を実行します。

1. 1つ以上のステップを選択します。
2. 右クリックして「**Move Into New SubPlan**」を選択します。

Design Studio によって、ステップが新しいサブプランにグループ化されます。

3. 必要に応じて、新しいサブプランの入 / 出力間のコネクタ線を追加または移動します。

サブプランのステップのグループ解除

サブプランのステップをグループ解除し、ステップをデータ・フローのレベルに戻すことができます。Design Studio によって、サブプランに存在する同じ接続点とコネクタ線を使用してデータ・フローが接続されます。反復を有効にしているサブプランをグループ解除すると、反復設定はプランから削除されます。

- サブプランを右クリックし、「**Replace SubPlan with Steps It Contains**」を選択します。

データ・フローには、サブプランのかわりにステップが表示されます。

サブプランの移動

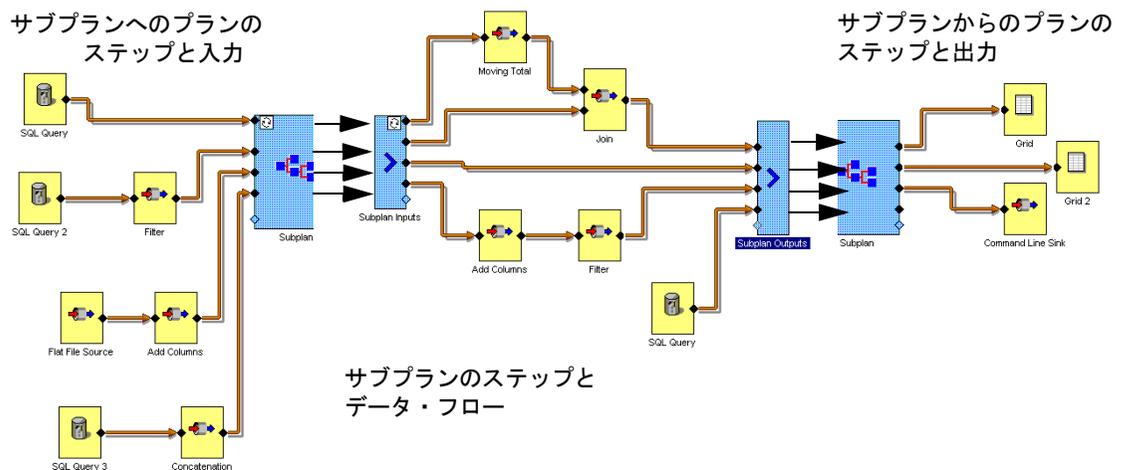
プランのレベル間を移動することができます。

サブプランで1つ上のレベルに移動するには、次の手順を実行します。

- サブプランで右クリックし、「**Show Containing SubPlan**」を選択します。

サブプランでの作業

サブプランでのデータ・フローの作成方法は、メイン・レベルのプランでの作成方法と同じです。Design Studio によって、新しい入力接続点と出力接続点が、必要に応じてサブプラン入力とサブプラン出力に追加されます。データ・フローのコネクタ線を接続点のいずれかに接続するたびに、Design Studio によって接続点が追加されます。



新しいサブプランを追加し、サブプランにデータ・フロー・コネクタ線を追加した後、そのサブプランを初めて表示すると、サブプラン入力記号およびサブプラン出力記号に次のものが示されます。

- サブプランの反復入力接続点に対応する反復出力接続点（接続されている場合）
- 各サブプランの入力データ・フローに対応する出力接続点
- 各サブプランの出力データ・フローに対応する入力接続点

サブプランおよびサブプランの接続点では、処理の結合、マージまたは分割は実行されません。サブプラン内の各接続点は、サブプランに接続されているデータ・フローの継続です。プランが正常に動作するために、サブプラン内部の各接続点は、データ・フローに接続されている必要があります。ただし、正常に動作するために入力ソースのみが必要なサブプラン内部の反復接続点は除きます。

サブプラン内の **Join Transform**、**Union Transform** および **Splitter Transform** を使用して、データ・フローの数を増加または削減できます。また、非表示シンクを使用して、サブプラン内のデータ・フローを終了できます。反復が有効な場合は、表示シンクを使用して、サブプラン内のデータ・フローを終了できます。ただし、表示シンクに表示されるのは、最初の反復からのデータのみです。

サブプランにデータ・フローを作成するには、次の手順を実行します。

1. サブプラン入力とサブプラン出力間で **Transform** をドラッグ・アンド・ドロップします。

Design Studio によって、ステップが追加され、コネクタ線がサブプラン入力記号の最初のサブプラン出力接続点と、サブプラン出力記号の最初のサブプラン入力接続点に自動的に追加されます。

2. データ・フローの反復を有効にするには、サブプラン入力コネクタ線をクリックして、**[Delete]** キーを押します。サブプラン入力の反復接続点から反復対象のデータ・フローにおける最初の **Transform** の入力まで新しいコネクタ線を引きます。

データ・フローのコネクタ線は、反復プロパティを設定するために、プラン・レベルでサブプランに接続する必要があります。通常、サブプラン入力ステップでの反復コネクタは、キー・フィールドによる反復の実行時に使用されます。サブプランでの反復プロパティの使用を目的とした場合、データ・フローはサブプラン内部の反復接続点に接続する必要はありません。

3. 反復接続点が無効な場合は、反復プロパティを設定します。

詳細は、[96 ページの「サブプランにおける反復プロパティの設定」](#)を参照してください。

4. すべてのステップを完成させ、テストするまで、サブプラン・ステップとコネクタの追加を続行します。
5. サブプランの入力または出力に未接続のものがいないことを確認します。

データ・フローからのサブプランの削除

サブプランを削除するには、次のいずれかを実行します。

- サブプランを右クリックし、「**Transform**」メニューで「**Delete**」を選択します。サブプランと、そこに含まれるステップが削除されます。
- サブプランをクリックし、**[Delete]** キーを押します。サブプランと、そこに含まれるステップが削除されます。

サブプラン内のステップを削除することなくサブプランを削除する方法については、[93 ページの「サブプランのステップのグループ解除」](#)を参照してください。

プランでの反復の使用

サブプラン記号には、反復接続点が含まれています。データ・フローを反復接続点に追加した後、反復プロパティを定義できます。サブプランの反復では、レコードまたはキー・フィールド・グループのいずれかを使用できます。

ここで説明する内容は次のとおりです。

- [レコードによる反復](#)
- [キー・フィールド・グループの反復](#)
- [複数のデータ・フローとサブプラン](#)
- [サブプランにおける反復プロパティの設定](#)
- [例:データ・セットにおける反復の効果](#)
- [例:レコード・グループによる条件付き反復](#)

レコードによる反復

このタイプの反復では、サブプランは、サブプランの前にあるステップからの処理変更を含む、データ・ソースからの最初のレコードを要求してから、サブプランに含まれるステップの操作の実行を開始します。1つのレコードがサブプラン処理を完了するたびに、その結果が出力され、新しいレコードが要求および処理されます。レコードでデータ処理を反復するサブプランは、データ・セット内の各レコードを新しいプランの実行として扱います。

このタイプの反復を使用し、データの各レコードを処理するプランを作成してから、レコードごとに出力します。たとえば、企業内の各店舗マネージャに対し、数に限りがある在庫品目を含む、未処理の注文を示すレポートを作成します。

キー・フィールド・グループの反復

このタイプの反復では、サブプランは、データ・ソースからソース・データを収集し、サブプランの前にあるステップを実行してから、選択されたキー・フィールドに応じてデータをソートします。データのソート後、レコード・グループで処理が開始されます。このレコード・グループは、ソート済データ・セットによって決定されます。レコード・グループでサブプラン処理が完了すると、サブプラン・ステップがリセットされ、新しいレコード・グループがソート済データ・セットのコレクションから選択されて処理されます。キー・フィールド・グループによるデータ処理の反復を行うサブプランは、収集されたデータ・セットにおける各レコード・グループを、新しいプランの実行として扱います。

このタイプの反復を使用し、レコード・グループを処理するプランを作成してから、レコード・グループごとに出力します。たとえば、企業内の各地区マネージャに対し、数に限りがある在庫品目を含む、未処理の注文を示すレポートを、地区、州、または他のグループごとに作成します。

サブプランに反復を設定する方法については、[96 ページの「サブプランにおける反復プロパティの設定」](#)を参照してください。

複数のデータ・フローとサブプラン

1つのサブプランには複数の入力を設定できますが、反復接続点に接続できるのは、1つの入力のみです。サブプランの処理中、非反復入力は、バッファ済の同じデータ・セットを反復ごとにサブプランにフィードします。これは、データの最初のブロックから行われます。Design Studio では、サブプランに存在するデータ・フロー制御のタイプ（結合反復データ・フロー、非反復データ・フローなど）に応じて、非反復データ・セットが処理されます。

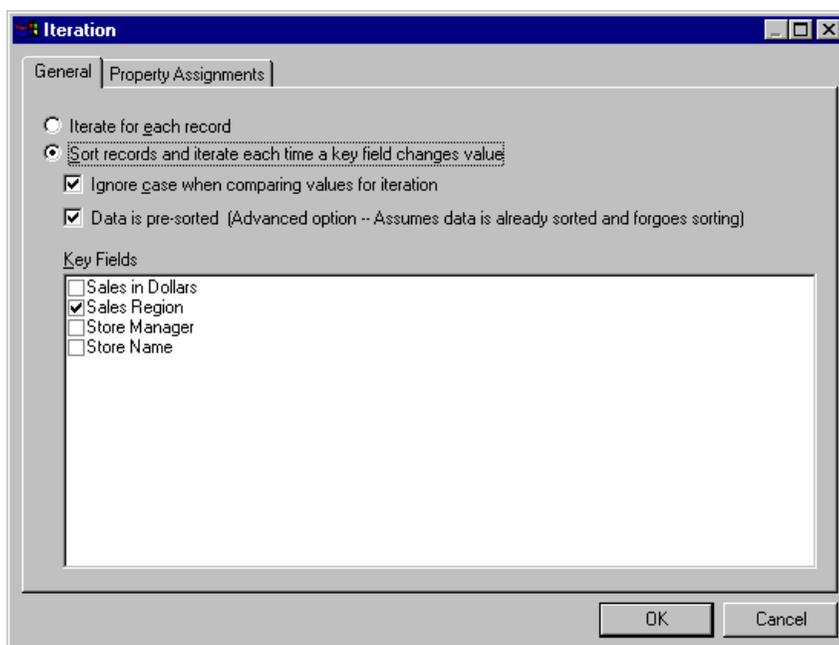
サブプランにおける反復プロパティの設定

サブプランの反復設定を編集するには、サブプランの反復接続点に接続されているデータ・ソース Transform が含まれているデータ・フローが必要です。

サブプランの反復設定を編集するには、次の手順を実行します。

1. サブプラン記号を右クリックし、「Transform」メニューから「Iteration」を選択します。

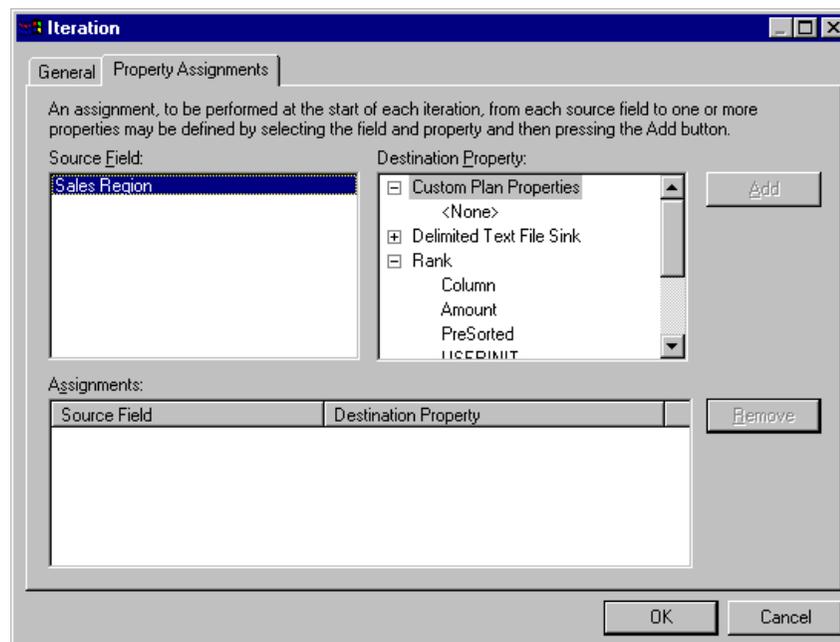
「SubPlan Iteration」ダイアログ・ボックスが表示されます。



2. 次の設定のいずれかを選択します。

- データ・ソースの各レコードを処理するには、「**Iterate For Each Record**」を選択します。
- 選択したキー・フィールドに基づいてレコードをソートした後、値が変更されるたびにグループのレコードを処理するには、「**Sort Records and Iterate Each Time A Key Field Changes Value**」を選択します。

3. キー・フィールドでこのサブプランを反復するように選択した場合は、次の手順を実行します。
 - a. 「Key Fields」リストから1つ以上のフィールドを選択します。
 「Key Fields」リストには、データ・ソース・ステップに追加したすべての Part が表示されます。使用するフィールドがリストに表示されていない場合は、データ・ソースの「Step Properties」ダイアログ・ボックスに移動して Part を追加してから、「SubPlan Iteration」ダイアログ・ボックスに戻ります。
 - b. Design Studio でキー・フィールドのレコード・グループ値を比較する際に大 / 小文字を区別しない場合は、「Ignore Case When Comparing Values For Iteration」を選択します。
 - c. データ・フローに含まれているデータが（先行するソート Transform などによって）すでにソートされている場合は、反復処理時に不要なソートが行われないように「Data is Pre-Sorted」を選択します。
4. 「Property Assignments」タブをクリックします。
 「Property Assignments」タブが表示されます。



5. ソース・フィールドとカスタム・プラン・プロパティ間の割当てを作成します。
 - a. 「Source Field」リストからソース・フィールドを選択します。
 - b. 「Destination Property」リストからカスタム・プラン・プロパティまたはステップ・プロパティを選択します。
 - c. 「Add」をクリックします。
 Design Studio によって、「Assignments」リストに、ソースと割当て先が表示されます。
6. 「Continue」をクリックします。

例：レコード・グループによる条件付き反復

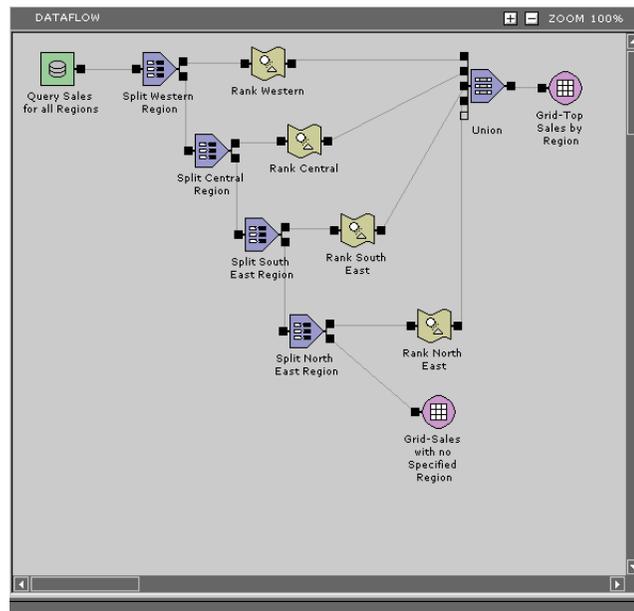
通常では複数のステップおよびデータ・パスを必要とする、データ抽出タスクおよびデータ変換タスクの多くは、サブプラン Transform および反復処理を使用すると、少ないステップで実行できます。

この例では、上位 5 つの売上高を地区ごとに生成し、それを Analysis 表示で表示するプランを作成する必要があります。結果の Analysis 表示から、クロス集計およびグラフを書式設定できます。

選択できる方法は、次の 2 つです。

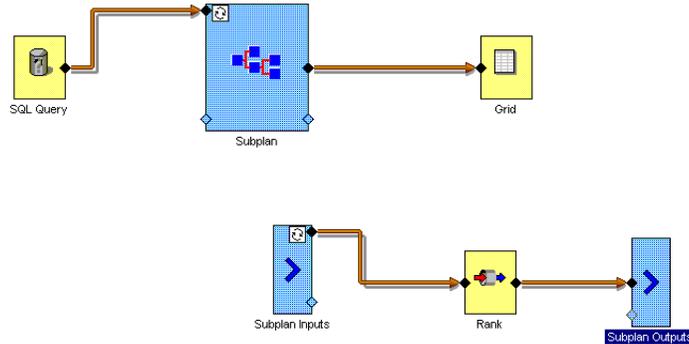
- 方法1: 出力を 4 つの Sales Region で分割し、Sales in Dollars でランク付けしてから、結果を単一の表示にマージします。

この方法を使用する場合、各販売地区の上位 5 つの売上高のみが昇順で表示されるように、出力を分割し、それぞれの結果のデータ・フローをランク付けします。Union を使用して各ランキングの結果を共通の結果セットにマージし、その結果を Grid 表示または Analysis 表示で表示します。



- 方法2: Sales Region を使用して Key Field Group で反復し、Sales in Dollars でランク付けしてから、表示する結果を出力します。

この方法を使用する場合、問合せ入力をサブプランに接続し、Sales Region Part を使用して反復するようにサブプランを設定してから、Sales in Dollar に基づいて出力をランク付けします。



どちらの方法でも同じ結果セットが生成されますが、サブプランを使用する方が作成時間が短く、理解も容易です。

例：データ・セットにおける反復の効果

次の表に、この例で使用するサンプル・データ・セットを示します。

Store #	City	State	Order #	Order Date	Allocated Item
005	Norman	OK	OK-496850	01/21/02	AB-8563
007	Tulsa	OK	OK-596831	01/22/02	AB-4593
009	Lawrence	KS	KS-305968	01/21/02	BC-4950
011	Topeka	KS	KS-294056	01/22/02	DF-5948
013	Lincoln	NE	NE-493018	01/21/02	RT-4503
015	Omaha	NE	NE-512948	02/23/02	BD-4950
017	Aberdeen	SD	SD-500794	01/21/02	BC-4950
019	Sioux Falls	SD	SD-124023	01/21/02	AB-8563
021	Bismarck	ND	ND-690506	01/21/02	RT-4503
023	Fargo	ND	ND-192045	01/21/02	AB-8563

ケース 1: 1 つのキー・フィールドによる反復

State をキー・フィールドとして指定すると、Data Flow Service によって、レコードの選択と、このフィールドに基づくレコードのソートが行われた後、Design Studio によって、各レコード・グループに対し、サブプランに含まれるステップが実行されます。このケースでは、データ入力は、次のレコード・グループで処理されます。

レコード・グループ	Store #	City	State	Order #	Order Date	Allocated Item
1	009	Lawrence	KS	KS-305968	01/21/02	BC-4950
	011	Topeka	KS	KS-294056	01/22/02	DF-5948
2	021	Bismarck	ND	ND-690506	01/21/02	RT-4503
	023	Fargo	ND	ND-192045	01/21/02	AB-8563
3	013	Lincoln	NE	NE-493018	01/21/02	RT-4503
	015	Omaha	NE	NE-512948	02/23/02	BD-4950
4	005	Norman	OK	OK-496850	01/21/02	AB-8563
	007	Tulsa	OK	OK-596831	01/22/02	AB-4593
5	017	Aberdeen	SD	SD-500794	01/21/02	BC-4950
	019	Sioux Falls	SD	SD-124023	01/21/02	AB-8563

ケース 2: 2 つのキー・フィールドによる反復

Allocated Item と Order Date をキー・フィールドとして指定すると、Data Flow Service によって、レコードの選択と、これらのフィールドに基づくレコードのソートが行われた後、Design Studio によって、各レコード・グループに対し、サブプランに含まれるステップが実行されます。このケースでは、データ入力は、次のレコード・グループで処理されます。

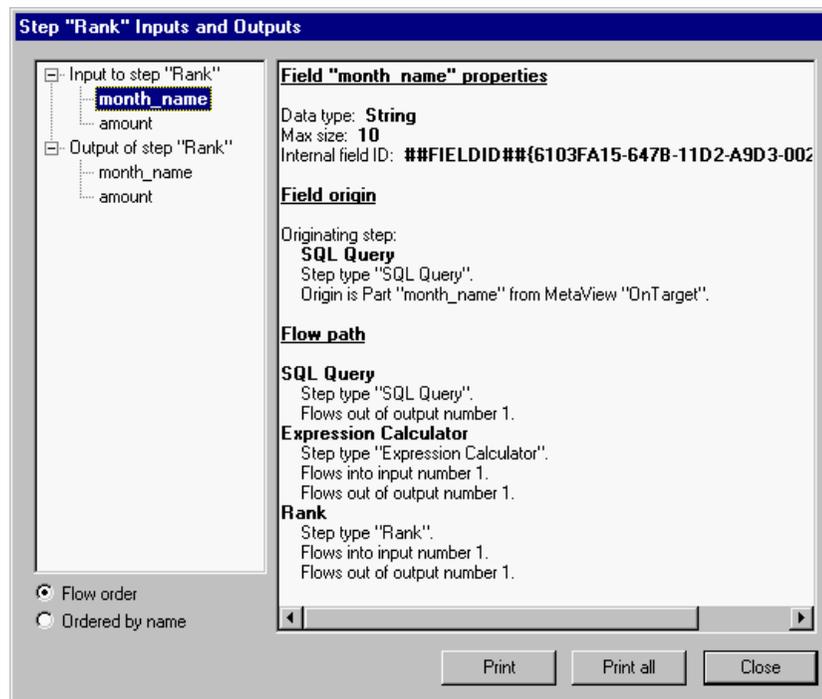
レコード・グループ	Store #	City	State	Order #	Order Date	Allocated Item
1	007	Tulsa	OK	OK-596831	01/22/02	AB-4593
2	005	Norman	OK	OK-496850	01/21/02	AB-8563
	019	Sioux Falls	SD	SD-124023	01/21/02	AB-8563
	023	Fargo	ND	ND-192045	01/21/02	AB-8563
3	009	Lawrence	KS	KS-305968	01/21/02	BC-4950
	017	Aberdeen	SD	SD-500794	01/21/02	BC-4950
4	015	Omaha	NE	NE-512948	02/23/02	BD-4950
5	011	Topeka	KS	KS-294056	01/22/02	DF-5948
6	013	Lincoln	NE	NE-493018	01/21/02	RT-4503
	021	Bismarck	ND	ND-690506	01/21/02	RT-4503

ステップの入力と出力の表示

プラン内のステップの入力および出力フィールドを表示できます。この情報は、入力フィールドの開始点を特定したり、データ・フロー内のパスをトレースする場合に役立ちます。

入力および出力フィールドを表示するには、次の手順を実行します。

- ▶ データ・フローのステップを右クリックし、「Inputs and outputs」を選択します。



「Inputs and Outputs」ダイアログ・ボックスには、各入力フィールドのプロファイルと、データ・フロー内のパスが示されます。左側のツリーでアイテムをクリックし、右側のパネルに詳細を表示します。

プロファイル情報には、次の情報が含まれています。

セクション	説明
Field properties	列タイプ、サイズ、およびリポジトリにおける一意のオブジェクト ID を示します。
Field origin	このフィールドが追加されたデータ・フロー内のステップ、オリジナルの Part および MetaView（存在する場合）を示します。
Flow path	このフィールドが通過するデータ・フロー内のすべてのステップを示します。

デフォルトでは、左側のフィールド・ツリーは、データ・フロー内の列の順序を反映しています。また、フィールド名でアルファベット順にリストをソートすることもできます。

フィールド・リストをアルファベット順にソートするには、次の手順を実行します。

➤ 「**Ordered by name**」 ボタンをクリックします。

現行のステップの入力または出力フィールドをすべて表示できます。入力または出力フィールドを表示するには、次の手順を実行します。

➤ 「**Input to step**」 または 「**Output of step**」 をクリックします。

ダイアログ・ボックスには、現行のステップのすべての入力または出力フィールドのリストが表示されます。名前はデータ・フローに表示されるとおりに表示され、フィールド・タイプはカッコ内に表示されます。

また、ステップまたはフィールドの現行のプロファイルを印刷することもできます。選択項目の情報を印刷するには、「**Print**」 をクリックします。ステップの情報をすべて印刷するには、「**Print All**」 をクリックします。

注意：「**Print All**」 を選択した場合、それぞれの情報セットが別々のページに印刷されます。情報セットを1つのページにまとめるには、**Design Studio** からテキストをコピーし、それをテキスト・ファイルに貼り付けてから、そのファイルを印刷します。

Transform の共有

Transform は、あるリポジトリから別のリポジトリにエクスポートおよびインポートすることで、他のユーザーと共有できます。エクスポートおよびインポートは、すでに指定された設定があるカスタム Transform を共有する場合に役立ちます。また、Transform Software Developer's Kit (SDK) を使用して C++ で記述されたカスタム Transform を配布する場合にも役立ちます。Transform SDK の詳細は、『Transform Programmer's Guide』を参照してください。

ここで説明する内容は次のとおりです。

- [Transform のエクスポート](#)
- [Transform のインポート](#)

Transform のエクスポート

Transform をエクスポートすると、その Transform 定義はローカル・ファイルまたはネットワーク・ファイルに格納されます。その後で、Transform を別のリポジトリにインポートすると、そのリポジトリのユーザーがアクセスできるようになります。

Transform をエクスポートするには、次の手順を実行します。

1. Tool Bin で Transform を右クリックし、「Export」を選択します。

「Export Tool」ダイアログ・ボックスが表示されます。

2. TFM ファイルの名前を入力し、場所を指定します。

デフォルトのディレクトリは、Oracle BAM Enterprise Link プログラム・ファイルが格納されているフォルダです。

Transform をインポートするユーザーが、使用しているローカル・ネットワークに存在していない場合は、そのユーザーがアクセスできるフォルダにファイルを保存します。これができない場合は、電子メールや他の方法を使用して、ファイルを転送することもできます。

3. 「OK」をクリックします。

Transform 定義が現行のリポジトリからエクスポートされ、ファイルに格納されます。

Transform のインポート

別のリポジトリからエクスポートされた Transform をインポートできます。インポートすると、Transform ファイルが、Transform として Tool Bin に追加されます。

Transform をインポートするには、次の手順を実行します。

1. Tool Bin で右クリックし、「Import」を選択します。

「Import Tool」ダイアログ・ボックスが表示されます。

2. Transform ファイルが自分の Oracle BAM Enterprise Link プログラム・フォルダに存在する場合は、それをダブルクリックします。それ以外の場合は、ファイルの場所を参照します。

3. 「OK」をクリックします。

Transform が現行のリポジトリにインポートされ、Tool Bin に表示されます。

Tool Bin 内の各 Transform には、現行のリポジトリで一意的オブジェクト ID があり、この ID は、他の情報とともに TFM ファイルに格納されます。使用しているリポジトリにこの ID を持つ Transform がすでに含まれている場合、インポートされる Transform には新しいオブジェクト ID が指定されます。

注意： C++ で記述されたカスタム Transform をインポートすると、オブジェクト定義のみがコピーされます。また、使用しているシステムが Transform DLL にアクセスできることを確認し、その Transform を登録する必要もあります。詳細は、『Oracle Business Activity Monitoring Enterprise Link 管理者ガイド』を参照してください。

プロパティでの作業

ここで説明する内容は次のとおりです。

- 概要
- Transform プロパティの使用
- カスタム・プラン・プロパティの使用
- プラン・プロンプトの使用

概要

プロパティはデータ・フローに設定できるプランのパラメータです。これらのパラメータは、**Design Studio** および他のアプリケーション環境でプランをカスタマイズする場合に有効です。プロパティの実際の使用例は次のとおりです。

- プランの結果のフィルタ
- プランの実行時のエンド・ユーザーへの値の入力の要求
- **WebLink** または **SARUN** でのプランの実行時のプランのカスタマイズ

プランが作成されると、作成されたプランには **Transform** プロパティの基本セットが含まれています。**Transform** プロパティは事前定義され、**Transform** のダイアログ・ボックスのフィールドに対応します。ダイアログ・ボックスのフィールドを追加または削除できないため、このタイプのプロパティは追加または削除できません。**Transform** プロパティは、プランの1つのステップのみに影響します。

プランが作成されると、カスタム・プラン・プロパティを追加できます。カスタム・プラン・プロパティは、定義するパラメータです。**Transform** プロパティとは異なり、必要に応じてカスタム・プラン・プロパティを追加および削除できます。カスタム・プラン・プロパティは、プランのすべてのステップに影響します。

Transform プロパティおよびカスタム・プラン・プロパティの両方は、割当て済のデータ型をもつ名前 / 値ペアです。たとえば、**Rank Transform** では、ランクに対する列は、ユーザー割当ての値およびデータ型文字列を含む列プロパティによって決定されます。サポートされる型のリストについては、[111 ページ](#)の「サポートされている型」を参照してください。

Transform プロパティの使用

Transform プロパティは、**Transform** のダイアログ・ボックスのフィールドに対応する事前定義されたプロパティです。それらの値をもつ **Transform** プロパティは、設定と呼ばれます。「**Step Settings**」ダイアログ・ボックスおよび **Transform** のダイアログ・ボックスの両方で設定を表示および変更できます。

ここで説明する内容は次のとおりです。

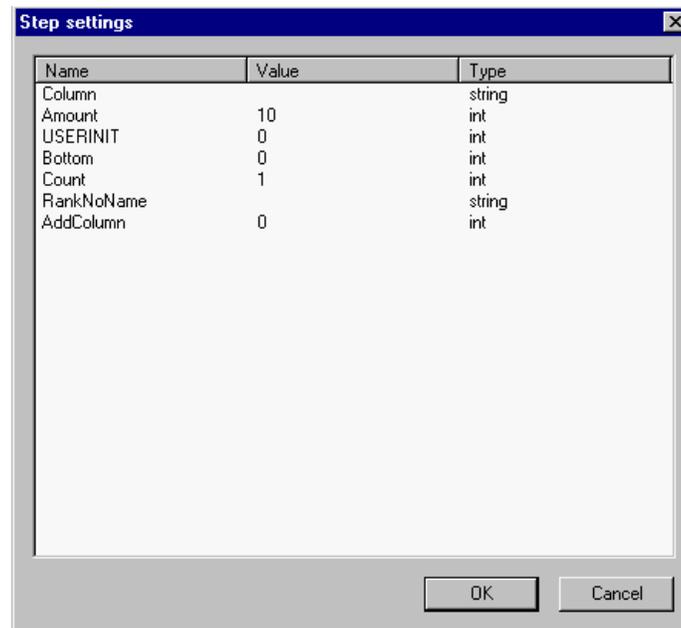
- 「**Step Settings**」の表示
- **USERINIT** の設定

「Step Settings」の表示

Transform のダイアログ・ボックスでは、プロパティの説明キャプションが提供され、値が検証されます。「Step Settings」ダイアログ・ボックスでは、変更できる Transform プロパティの名前、値および型が表示されます。プロパティは、削除または追加できません。

Transform の「Settings」ダイアログ・ボックスを開くには次のようにします。

➤ データ・フローの Transform を右クリックし、「Settings」を選択します。



プロパティの値を変更するには、次の手順を実行します。

1. 「Settings」ダイアログ・ボックスの値をクリックします。
2. 値を編集するには、再度クリックします。

プランに対してカスタム・プラン・プロパティが定義される場合、「Value」フィールドはドロップダウン・リストになります。このリストからカスタム・プラン・プロパティを選択して、それを Transform プロパティに割り当てます。カスタム・プラン・プロパティの詳細は、[108 ページの「カスタム・プラン・プロパティの使用」](#)を参照してください。

3. 「OK」をクリックして、値を適用します。

このダイアログ・ボックスに設定する値は、Transform のダイアログ・ボックスに表示されます。ステップ・プロパティの詳細は、[90 ページの「ステップのプロパティの編集」](#)を参照してください。

USERINIT の設定

すべての Transform には、integer 型の USERINIT プロパティがあります。このプロパティは、Transform がユーザーからアクセスされたかどうかを監視します。Transform のダイアログ・ボックスを開いていない場合、USERINIT の値は 0 に設定されます。Transform のダイアログ・ボックスを開き、「OK」をクリックして閉じる場合、USERINIT の値は 1 に変更されます。

USERINIT は、Transform の他のすべてのプロパティを有効にするフラグです。Transform のダイアログ・ボックスを使用して、Design Studio でプロパティを設定すると、USERINIT の値は変更されます。ただし、ダイアログ・ボックスが開かれていない場合、手動で USERINIT を 1 に変更する必要があります。たとえば、「Step Settings」ダイアログ・ボックスを使用して、Rank Transform のすべてのプロパティを設定する場合、USERINIT を 1 に設定して、ランクに対する列の名前など、他のプロパティの値を有効にします。

USERINIT を 0 に設定する場合、プランの実行時に Transform のすべてのプロパティが無視されます。データは Transform を介して、変更されていない次のステップまで移動します。プランの動作に関する影響は、Transform をプランから削除する場合に類似しています。

カスタム・プラン・プロパティの使用

カスタム・プラン・プロパティは、Transform プロパティに類似したパラメータです。違いは、カスタム・プラン・プロパティでは名前、値および型を定義することです。カスタム・プラン・プロパティを作成すると、プロパティをサポートするプランのすべてのステップで、入力としてプロパティを使用できます。

計算 Part において、および SQL 文の置換文字列として、プラン・プロンプトでカスタム・プラン・プロパティを使用できます。所有するプランに対してカスタム・プラン・プロパティを作成、変更および削除できます。

Design Studio におけるカスタム・プラン・プロパティの使用例も提供されません。

ここで説明する内容は次のとおりです。

- [カスタム・プラン・プロパティの表示](#)
- [カスタム・プラン・プロパティでの作業](#)

カスタム・プラン・プロパティの表示

カスタム・プラン・プロパティは、「Plan Properties」ダイアログ・ボックスに表示されます。すべてのプランのカスタム・プラン・プロパティを表示できます。

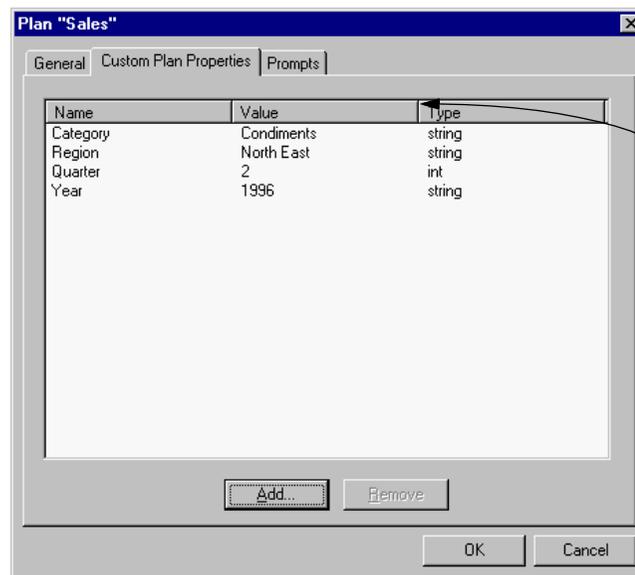
カスタム・プラン・プロパティを表示するには、次の手順を実行します。

1. 「Tools」 → 「Plan Properties」 を選択します。

現行のプランの「Properties」ダイアログ・ボックスが表示されます。Plan Bin 内のプランを右クリックし、「Properties」を選択して、「Properties」ダイアログ・ボックスを開くこともできます。

2. 「Properties」ウィンドウの「Custom Plan Properties」タブをクリックします。

タブには、現行のプランに対して定義されたカスタム・プラン・プロパティが表示されます。



列をサイズ変更するには、
ここをドラック

カスタム・プラン・プロパティでの作業

「Plan Properties」ダイアログ・ボックスのカスタム・プラン・プロパティを追加および変更します。所有するすべてのプランに、カスタム・プロパティを追加できます。

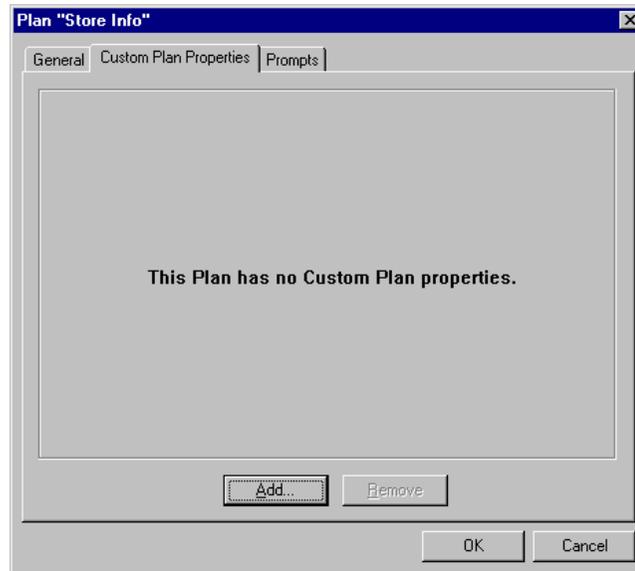
カスタム・プロパティをプランに追加するには、次の手順を実行します。

1. データ・フロー・エディタで右クリックして、「Plan Properties」を選択します。

ダイアログ・ボックスでは、このプランに関する一般的な説明が表示されます。

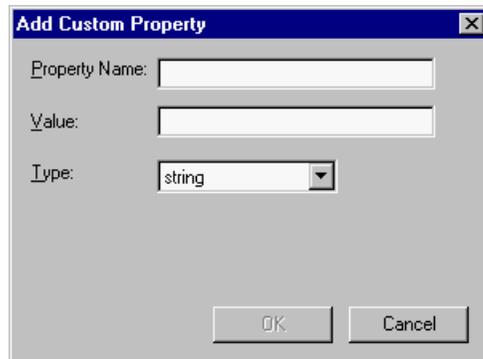
2. 「Custom Plan Properties」 タブをクリックします。

カスタム・プラン・プロパティが定義されていない場合、ダイアログ・ボックスにはこのプランのプロパティがないことが示されます。



3. 「Add」 ボタンをクリックします。

「Add Custom Property」 ダイアログ・ボックスが表示されます。



4. 「Property Name」 フィールドに、プロパティの名前を入力します。

5. 「Value」 フィールドに、プロパティの値を入力します。

6. ドロップダウン・リストからデータ型を選択します。

サポートされている型	説明
string	文字列
string list	カンマで区切られた文字列のリスト
int	-2,147,483,648 から 2,147,483,647 の範囲の 4 バイトの整数
int list	カンマで区切られた -2,147,483,648 から 2,147,483,647 の範囲の 4 バイトの整数のリスト
long	-2,147,483,648 から 2,147,483,647 の範囲の 4 バイトの整数
long list	カンマで区切られた -2,147,483,648 から 2,147,483,647 の範囲の 4 バイトの整数のリスト
datetime	実数としての日付と時刻
datetime list	カンマで区切られた実数としての日付と時刻のリスト
double	64 ビット数値としての倍精度の浮動小数点数
double list	カンマで区切られた 64 ビット数値としての倍精度の浮動小数点数のリスト
uint	符号なし整数
uint list	カンマで区切られた符号なし整数のリスト
decimal	小数点を含み、負数の符号を付けられる数値。精度とスケールは、両方とも整数で最大 38 桁に制限されます。
decimal list	カンマで区切られ、小数点を含み、負数の符号を付けられる数値リスト。精度とスケールは、両方とも整数で最大 38 桁に制限されます。

7. 「OK」をクリックします。

カスタム・プラン・プロパティはプランに追加されます。ユーザーがプロパティ・データ型と互換性のない値を指定すると、プロパティにはデフォルト値が割り当てられます。たとえば、プロパティのデータ型に `int` を選択して、`string` 値を指定すると、プロパティにはデフォルトの `int` 値 `0` が割り当てられます。

注意： カスタム・プラン・プロパティは変更できません。かわりに、プロパティを削除し、新しい名前の新しいプロパティを追加します。

プランからカスタム・プロパティを削除するには、次の手順を実行します。

1. 削除するプロパティの名前をクリックします。
2. 「Remove」をクリックします。
3. 「OK」をクリックします。

現行のプランからプロパティが削除されます。このプロパティを参照するプランを実行しようとする、プロパティが欠落していることがメッセージで警告されます。

既存のカスタム・プラン・プロパティの値を変更するには、次の手順を実行します。

1. 変更するプロパティの値フィールドをダブルクリックします。
2. プロパティの新しい値を入力します。
3. 「OK」をクリックします。

既存のカスタム・プラン・プロパティの型を変更するには、次の手順を実行します。

1. カスタム・プラン・プロパティの「Type」フィールドをクリックします。
2. ドロップダウン・リストから新しい型を選択します。
3. 必要に応じて「Value」フィールドに、プロパティの新しい値を入力します。

注意： カスタム・プラン・プロパティの値は、ともに使用される列の型と互換性がある必要があります。互換性がないと、プランの実行時に型変換エラーが発生します。これを解決するには、プロパティの新しい値を指定するか、または値を変換する SQL を変更します。

カスタム・プラン・プロパティの複数の値の選択

カスタム・プラン・プロパティでは、プロパティ値として複数のアイテムを使用できます。これは、ユーザーがプラン・プロンプトの基準として複数のアイテムを選択できることを意味します。

値リストを使用するカスタム・プラン・プロパティを設定するには、プロパティのリスト・データ型を選択します。たとえば、string データ型のかわりに、string list データ型を選択します。各データ型のリスト版は、「Add Custom Property」ダイアログ・ボックスにあります。その後、値リストを入力できます。

次の構文は、「Plan Properties」ダイアログ・ボックスでカスタム・プラン・プロパティの値を入力する場合とプラン・プロンプトでリストを入力する場合に使用されます。

- たとえば、次のようにリストの値はカンマによって区切られます。
123,4,-56,78.9.
- 一重引用符は2つの一重引用符で置換し、値全体を一重引用符で囲む必要があります。この例は次のとおりです。
'John''s Plan','Lisa''s','Joe''s'.
- カンマ、先頭空白、後続空白を含む値は、一重引用符で囲む必要があります。

プロパティのデータ型がリスト型でないときに値リストを入力すると、プロパティでは最初の値のみが使用されます。

また、カスタム・プラン・プロパティのすべての値または NULL 値のみも選択できます。カスタム・プラン・プロパティの作成では、特殊な値 <<ALL>> または <<NULL>> をプロパティ値に対して使用できます。

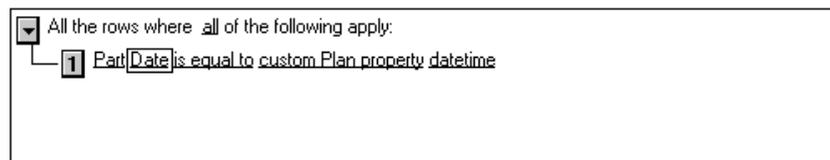
- <<ALL>> は、選択されたフィールドのすべての値を表します。文字列は、すべての値が返されるようにリクエスト時に SQL で変換されます。WHERE 句は、SQL 文から削除されます。
- <<NULL>> は、値が設定されなかったデータまたはレコードの NULL 値を表します。文字列は、レコード内のすべての NULL 値が選択されるようにリクエスト時に SQL で変換されます。

値リストを使用するカスタム・プラン・プロパティをプランに含めるには、プランにフィルタを追加する必要があります。「is in list」を選択して、フィルタにカスタム・プラン・プロパティを含めます。

複数の値のカスタム・プラン・プロパティを含むフィルタを作成するには、次の手順を実行します。

1. 次のようなプランのフィルタ式を作成します。
Part Sales Region is in list (empty string)
2. (empty string) というフィルタ式の最後のアイテムをクリックします。
「Filter List Editor」ダイアログ・ボックスが表示されます。
3. 値のタイプに「Custom Plan Property」を選択します。
4. 「Value」ドロップダウン・リストからカスタム・プラン・プロパティの名前を選択します。
5. 「Add New Value」をクリックして、値を追加します。
6. 「OK」をクリックしてダイアログ・ボックスを閉じます。

フィルタには、カスタム・プラン・プロパティに対して選択される複数の値が含まれます。



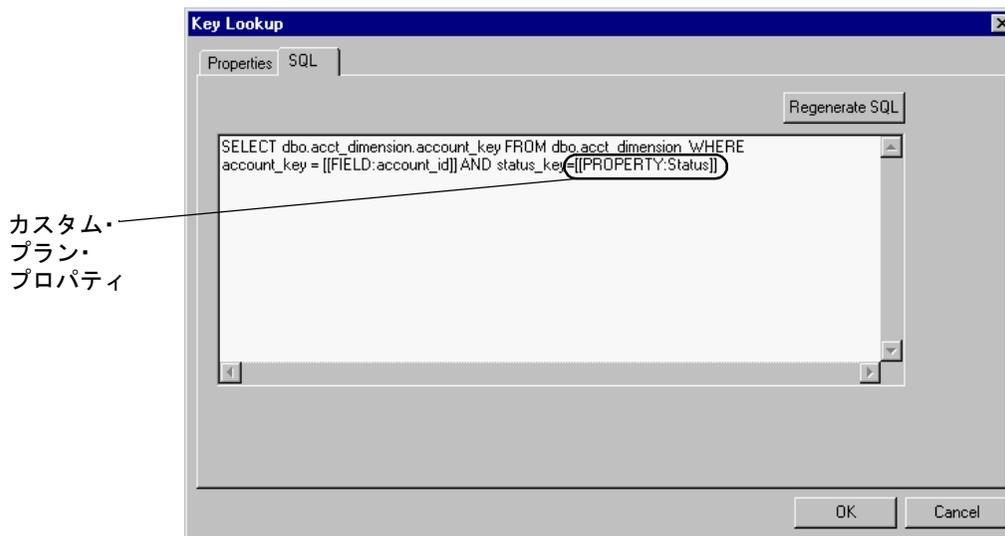
Transform のダイアログ・ボックスのカスタム・プラン・プロパティの使用

プロパティをもつすべての Transform について、Transform プロパティの値としてカスタム・プラン・プロパティを使用できます。手順については、107 ページの「[「Step Settings」の表示](#)」を参照してください。

いくつかの Transform では、Transform のダイアログ・ボックスのカスタム・プラン・プロパティを指定することもできます。この機能をもつ Transform は、SQL または式の構文を生成します。それらの Transform は次のとおりです。

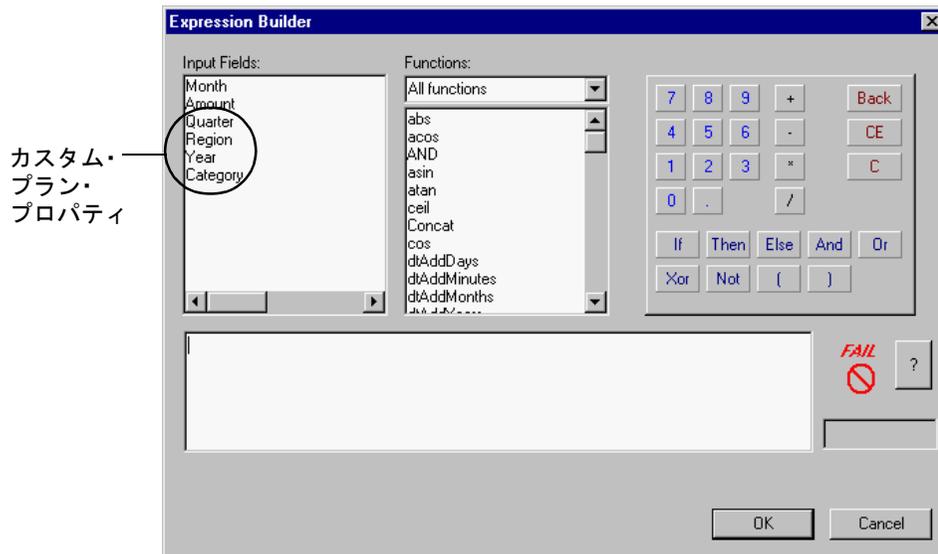
Transform	カスタム・プラン・プロパティの参照
SQL Query	SQL 文で手動入力
Key Lookup	SQL 文で手動入力
SQL Command	ドロップダウン・リスト
SQL Command Sink	ドロップダウン・リスト
Expression Calculator	入力フィールド
Query Iterator	リスト

SQL Query Transform および Key Lookup Transform の場合、SQL 文で直接、プロパティ構文を入力します。たとえば、次の SQL では、account_key 値を入力フィールド account_id と比較し、また、status_key が Status カスタム・プラン・プロパティの値と一致する必要があります。



このプロパティ構文を指定して、非アクティブなアカウント、またはクローズ状態のアカウントのファクトをロードする前に、アクティブなアカウントのファクト表をロードします。

他の Transform では、カスタム・プラン・プロパティは、Transform のダイアログ・ボックスの入力フィールド、またはドロップダウン・リストのオプションです。Expression Calculator Transform では、カスタム・プラン・プロパティは式で使用できる入力フィールドです。



「Expression Builder」ダイアログ・ボックスで、カスタム・プラン・プロパティの名前を入力またはドラックして式に追加します。プロパティ構文は必要ありません。

計算 Part のカスタム・プラン・プロパティの使用

Part 制約、および計算 Part 構文にカスタム・プラン・プロパティを含めることができます。両方の場合、制約はプランの SQL に追加されます。プロパティ構文および使用ルールは、Part 制約および計算 Part と同様です。

Part にカスタム・プラン・プロパティを指定する構文は次のとおりです。

```
[[Property:property_name]]
```

ここで、Property: is は静的テキストで、property_name はデータ・フロー内に適切に定義されたカスタム・プラン・プロパティの名前です。二重カッコは必須です。Property: is は大文字 / 小文字が区別されませんが、property_name を大文字にする場合、データ・フロー内の定義と正確に一致させる必要があります。

条件を追加するには、式で SQL 論理演算子を使用します。次に例を示します。

```
[[Property:state_prop]] = 'TN' OR [[Property:state_prop]] = 'CA'
```

この例では、状態値が TN または CA のいずれかの場合、フィールドを返します。

フィルタでのカスタム・プラン・プロパティの使用

プランのフィルタ条件として、カスタム・プラン・プロパティを使用できます。たとえば、プランには SQL Query ステップの次の制約が含まれる場合があります。

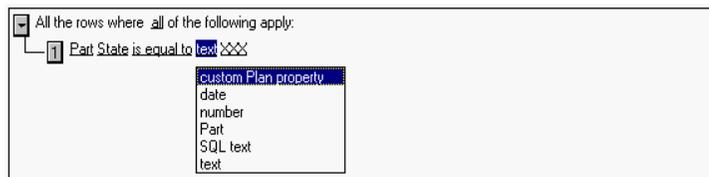
```
WHERE state = 'CA'
```

構文の変更を容易に行なうには、制約 CA をカスタム・プラン・プロパティに置き換えます。デフォルト値 CA を使用して、state_prop というプラン・プロパティを追加します。その後、フィルタを編集してこのプロパティを条件として使用します。制約構文は次のように変更されます。

```
WHERE State = [[PROPERTY: state_prop]]
```

カスタム・プラン・プロパティを使用してプランにフィルタを追加するには、次の手順を実行します。

1. 「**Show/Hide Filters**」 ボタンをクリックして、ワークスペースにフィルタを表示します。
2. フィルタの輪郭の下矢印をクリックして、新しい式エントリを挿入します。
3. フィルタで使用する Part および比較演算子を選択します。
4. 「**text**」 フィールドをクリックして、「**custom Plan property**」を選択します。



5. 次のフィールドで、使用するカスタム・プラン・プロパティを選択します。フィルタは SQL Query ステップに追加されます。

問合せを表示するには、次の手順を実行します。

1. データ・フロー・エディタ内の SQL Query ステップを右クリックして、「**Query Editor**」を選択します。
2. 「**SQL Viewer**」 タブをクリックします。

カスタム・プラン・プロパティは、WHERE 句の二重カッコに囲まれます。

プラン・フィルタの state_prop を使用して、ユーザーに値を求めるプラン・プロンプトを追加できます。ユーザーは、プラン内の定数のリストまたは列値リストから選択できます。

プランにおけるフィルタの使用の詳細は、オンライン・ヘルプを参照してください。

ユーザー入力の SQL でのカスタム・プラン・プロパティの使用

ユーザー入力の SQL で置換されたフィールドまたはカスタム・プラン・プロパティを使用できます。構文は、次のとおりです。

- `[[FIELD:field_name]]` (フィールドの場合)
- `[[PROPERTY:prop_name]]` (カスタム・プラン・プロパティの場合)

前述の構文を入力する場合、使用するデータベースのタイプに応じて、`[[FIELD?FMT=DB:field_name]]` と同等なデータベース書式設定のデフォルト値を指定します。

また、特定のデータベース書式設定タイプをフィールド値またはプロパティ値に適用するかどうかを制御することもできます。データベース書式設定を除外するには、次の構文を使用します。

`[[FIELD?FMT=NONE:field_name]]`

カスタム・プラン・プロパティ、計算 Part、次の Transform など、ユーザー入力の SQL が使用される場所でこの構文を使用できます。

- SQL Command
- SQL Query
- Key Lookup
- SQL Command Sink

フィールド値またはプロパティ値が NULL の場合、NONE 形式は適用できません。

プラン・プロンプトの使用

プラン・プロンプトは、プランの実行時にユーザーから入力を受け入れるダイアログ・ボックスです。ユーザーの入力は、表示するプランの結果に影響します。ユーザーがデータ・フローにアクセスせずにプロパティを設定できるように、プラン・プロンプトを追加できます。

プロパティの特定の値のみを受け入れるプロンプトを作成して、ユーザーのアクセスをプラン内のデータに制限できます。プラン・プロンプトは、プランとともに保存されます。また、作成するプランにプロンプトを追加するのと同様な方法で、サブスクライブするプランにプロンプトを追加することもできます。WebLink を介してプランを表示する場合、プラン・プロンプトは Web ページとして表示されます。VBA を含むプラン・プロンプトは、Web 上ではサポートされません。

ここで説明する内容は次のとおりです。

- [プラン・プロンプトの作成](#)
- [複数のパラメータの設定](#)
- [プラン・プロンプトの設定](#)
- [プラン・プロンプトの例の実行](#)

プラン・プロンプトの作成

プラン・プロンプトを作成するには、次の手順を実行します。

1. プロンプトを作成するプランを開きます。
2. 「Tools」 → 「Plan Properties」 を選択します。
3. 「Prompts」 タブをクリックして、「New Prompt」 ボタンをクリックします。
「Prompt Wizard」 ダイアログ・ボックスが表示されます。
4. ドロップダウン・リストからプロパティのソースを選択します。次のいずれかを選択できます。
 - 現行のプランのカスタム・プラン・プロパティ。カスタム・プラン・プロパティは、定義可能なパラメータです。カスタム・プラン・プロパティの作成の詳細は、[108 ページの「カスタム・プラン・プロパティの使用」](#)を参照してください。
 - 公開されているプロパティを含む、現行のプランのステップ。例としては、Rank Transform があります。公開されているプロパティをもつ Transform の詳細は、[107 ページの「Step Settings」の表示](#)を参照してください。

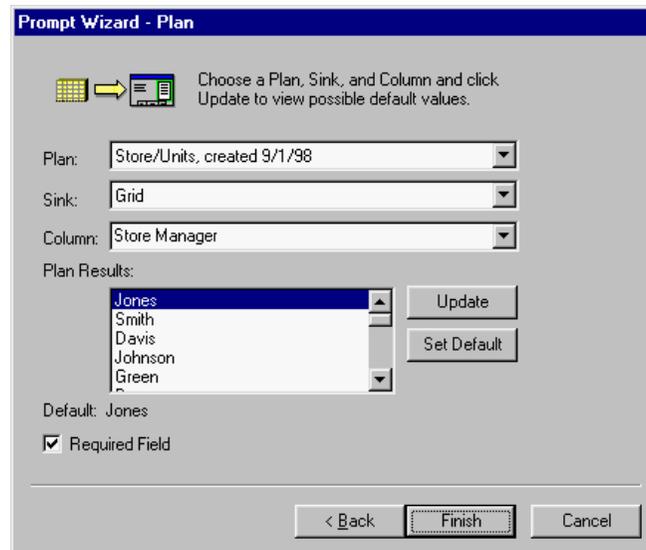
選択したソースに基づいて、プロパティのリストが表示されます。

5. 要求するプロパティを選択して、「Next」 をクリックします。
6. プラン・プロンプトに次の情報を入力します。
 - Title: プロンプト・タイトル・バーに表示されるテキスト。プロンプトの名前を追加できます。
 - Label: プロンプトのフィールドまたはリスト上に表示されるテキスト。プロンプトでの作業に関するユーザーの指示を指定できます。
 - Description: この「Description」 フィールドにのみ表示されるテキスト。自分自身または他の設計者に対する、プロンプトに関する注意事項を保存できます。

7. ドロップダウン・リストからユーザー入力のタイプを選択し、「Next」をクリックします。
8. 選択するユーザー入力のタイプに応じて、該当する手順に従います。

ユーザー入力	手順
Typed in	プロンプトのデフォルト値を入力します。この値は、プランの実行時にユーザーによる編集が可能なフィールドに表示されます。プロパティに日時データ型がある場合、編集可能なフィールドのかわりにカレンダーが表示されます。
Choose from a List	「Item」フィールドに値を入力して、「Add」をクリックします。リストにアイテムが追加されます。必要に応じてリストに他のアイテムを追加します。リスト内の値を選択し、値を並び替えまたは削除して、選択リストを作成します。プランの実行時にユーザーはこのリストから選択します。
Choose from a List of Plan Results	ドロップダウン・リストからプラン、シンク、列を選択し、「Update」ボタンをクリックします。結果には、プランの実行時にユーザーが選択するリストが作成されます。
Choose from a List of Distinct Part Values	ドロップダウン・リストから MetaView および Part を選択し、「Update」ボタンをクリックします。結果には、プランの実行時にユーザーが選択するリストが作成されます。

次の例は、プランの結果から生成されるリストです。



9. デフォルト値を指定するには、値を選択して「Set Default」をクリックします。
10. ユーザーがプラン実行前に選択する必要がある「Required Field」チェック・ボックスを選択します。
11. 「Finish」をクリックします。

「Prompts」タブ上のリストにプロンプトが追加されます。「Preview」をクリックすると、作成したプロンプトが表示できます。

複数のパラメータの設定

プラン・プロンプト・ウィザードには、値の複数の選択のサポートが含まれ、すべての値または NULL 値の選択がサポートされます。

プラン・プロンプトによって、プランの実行時にユーザーにパラメータを求めるプランを作成できます。これらのパラメータには、カスタム・プラン・プロパティまたは Transform プロパティの値が設定されます。プラン・プロンプトは、WebLink のユーザーに対しても表示されます。

プラン・プロンプト・ウィザードによって、すべての値、NULL 値または空白値を示すプロンプトに値を含めることができます。プラン・プロンプトにこれらの追加の選択を示すカスタム文字列を定義できます。ユーザーがプロンプトを変更しなかったり、すべての値を選択解除する場合に、結果を制御する設定を選択することもできます。

プランを実行してプロンプトを表示する場合、値の選択時に [Ctrl] キーまたは [Shift] キーを押して、プロンプト・リストの複数の値を選択できます。

次の状況では、複数の値を選択できる方法が決定されます。

- リスト・データ型であるカスタム・プラン・プロパティに対してプロンプトを定義する場合、プロンプトで値リストを選択できます。リスト、異なる Part 値およびプランの結果を表示するプロンプトの場合、プロンプトは複数の選択を受け入れます。
- 入力フィールドであるプロンプトの場合、正しい構文を使用して値リストを入力する必要があります。入力値には、**datetime** 値が含まれます。1つの **datetime** 値に対してのみカレンダー・コントロールが表示されます。
- 入力リストが正しい書式でないか、または選択された値のすべてがプロパティ・タイプに対して有効でない場合、プロンプトを続行することはできません。

次に、新機能を使用してプラン・プロンプトを設計する例を示します。この例では、<<ALL>> および <<NULL>> に対して新しい文字列が定義されますが、<<BLANK>> に対しては定義されません。値リストには、新しい文字列が表示されます。これらはプラン・プロンプトの実行時にユーザーが参照する値です。

Prompt Wizard - Distinct Part Values

Choose a MetaView and a Part, and click Update to view possible values. Set additional values.

MetaView: Accounting

Part: Phone, Sales Region, State, Store Manager

Distinct Part Values: North East, South East, Western, All Regions, No Region

Update, Set Initial Value

Initial value: <<ALL>>

Add these values to the list:

All values as: All Regions

Null value as: No Region

Blank or empty as: <<BLANK>>

User can leave value unselected:

Use current property value

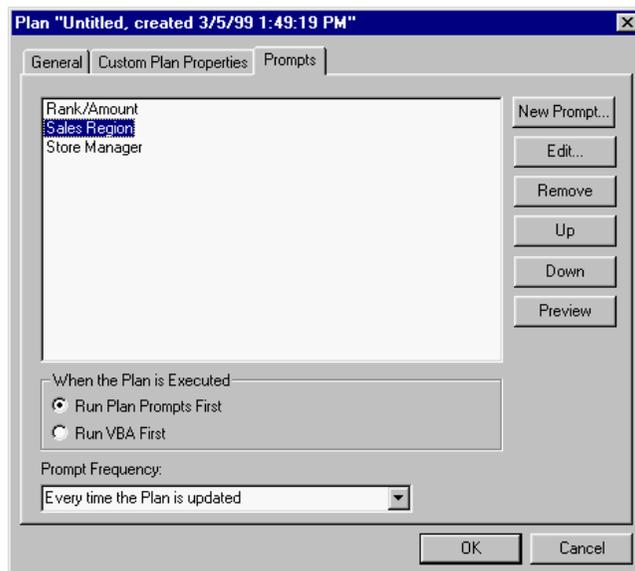
Set property to this value: <<ALL>>

< Back, Finish, Cancel

ALL および NULL 値は、カスタム・プラン・プロパティに基づいたプロンプトでのみ使用できます。Transform プロパティに <<ALL>> または <<NULL>> 値は選択できません。ALL 値はリスト・フィルタにのみ適用されます。また、空白値を示す文字列を定義することもできます。この文字列のデフォルト値は、<<BLANK>> です。

プラン・プロンプトの設定

「Plan Properties」ダイアログ・ボックスの「Prompts」タブを使用して、作成したプロンプトを編集、削除およびプレビューできます。プランに含まれる VBA コードの前または後でプロンプトの実行を選択して、プロンプトを実行する頻度を選択します。



ここで説明する内容は次のとおりです。

- [プラン・プロンプトの編集および削除](#)
- [プロンプトの表示順の変更](#)
- [プラン・プロンプトのプレビュー](#)
- [最初にプラン・プロンプトまたは VBA を実行](#)
- [プロンプトの頻度の選択](#)

次の表にボタンおよび設定の概要を示します。

アイテム	説明
「New Prompt」 ボタン	「Prompt Wizard」を使用して新しいプロンプトを追加します。
「Edit」 ボタン	「Prompt Wizard」を使用して、現在、選択されているプロンプトを編集します。
「Remove」 ボタン	現在、選択されているプロンプトを削除します。
「Up」 ボタン	現在、選択されているプロンプトをリストの上方に移動します。
「Down」 ボタン	現在、選択されているプロンプトをリストの下方に移動します。
「Preview」 ボタン	プランの実行またはプロパティの設定をしないで、すべてのプロンプトを表示します。
「When the Plan is Executed」 ラジオ・ボタン	「Run Plan Prompts First」または「Run VBA First」のいずれかを選択します。プランにプロンプトとVBAコードの組合せがある場合、その実行順を選択できます。デフォルトでは、最初にプラン・プロンプトを実行します。
「Prompt Frequency」 ドロップダウン・リスト	プロンプトの実行を、プランの初回更新時、プランの更新ごと、またはユーザーが「Run Plan Prompts」を選択したときのみ、から選択します。デフォルトでは、プランが実行されるたびにプロンプトが実行されます。

プラン・プロンプトの編集および削除

プラン・プロンプトを編集するには、次の手順を実行します。

1. 「Prompts」 タブで、プロンプト・リストのプロパティの名前を選択します。
2. 「Edit」 ボタンをクリックします。

「Prompt Wizard」 ダイアログ・ボックスが表示されます。タイトル、ラベルと説明、および値が編集できます。ユーザー入力タイプは変更できません。ユーザー入力タイプを変更するには、新しいプラン・プロンプトを作成します。プラン・ウィザードの詳細は、[125 ページの「プラン・プロンプトの例の実行」](#)を参照してください。

プラン・プロンプトを削除するには、次の手順を実行します。

1. プロンプト・リストのプロパティの名前を選択します。
2. 「Delete」 ボタンをクリックします。

プロンプトがリストから削除されます。

プロンプトの表示順の変更

プランが実行される際、プロンプトは、「Prompts」タブのリストに表示されている順で表示されます。プロンプトが表示される順序を変更できます。

プロンプトの順序を変更するには、次の操作を実行します。

1. 「Prompts」タブのリストでプロンプトを選択します。
2. 「Up」ボタンまたは「Down」ボタンをクリックして、リスト内のプロンプトの位置を変更します。

プラン・プロンプトのプレビュー

プラン・プロンプトをプレビューするには、次の手順を実行します。

1. 「Prompts」タブで、「Preview」ボタンをクリックします。
2. 複数のプロンプトがある場合は、「Next」をクリックし、最後のプロンプトで「Finish」をクリックして「Prompts」タブに戻ります。

このプランに関連付けられたすべてのプロンプトは、プランの実行時と同じように表示されます。プロンプトはプレビュー・モードであるため、プロンプトのダイアログ・ボックスで「Finish」をクリックすると、プランは実行されません。

最初にプラン・プロンプトまたは VBA を実行

プランにプロンプトおよび VBA を追加した場合、プロンプトを VBA の前に実行するか後に実行するかを選択できます。デフォルトでは、最初にプラン・プロンプトを実行します。

プラン・プロンプトの前に VBA を実行するには次のようにします。

- 「Prompts」タブで、「Run VBA First」ラジオ・ボタンをクリックします。

プロンプトの頻度の選択

プラン・プロンプトを実行する頻度を選択できます。ユーザーがメニュー・アイテムを選択する場合のみプロンプトを実行する必要がある場合があります。デフォルトでは、プランの更新ごとにプラン・プロンプトが実行されます。

プラン・プロンプトを実行する頻度を選択するには、次の手順を実行します。

- 「Prompt Frequency」ドロップダウン・リストから次のいずれかを選択します。
 - Every time the Plan is updated
 - The first time the Plan is updated
 - Only when the Run Plan Prompts menu item is selected。プロンプトを表示するには、ユーザーは「Tools」→「Run Plan Prompts」を選択する必要があります。

プラン・プロンプトの例の実行

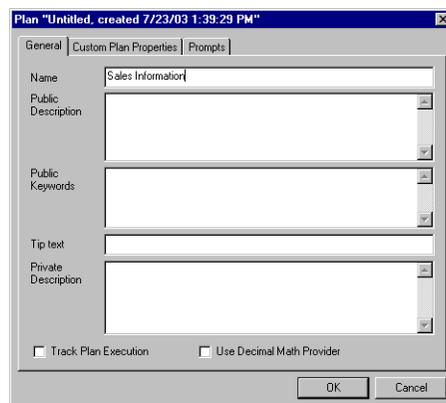
プラン・プロンプトを含むプランは、すべての他のプランと同じように実行します。「Update」をクリックすると、結果が表示される前に入力用のプロンプトが表示されます。次の例は、プランのプロンプトを作成して、プランを実行する方法を示しています。

datetime プロンプトを持つプランを作成してそのプランを実行するには、次の手順を実行します。

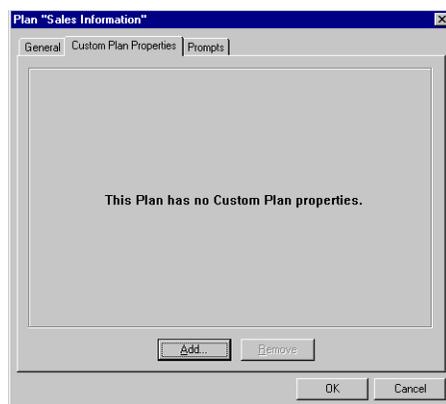
1. 「Store Manager」、「Sales Region」、「Sales in Dollars」および「Date」 Part をワークスペースにドラックします。

2. 「Tools」 → 「Plan Properties」 を選択します。

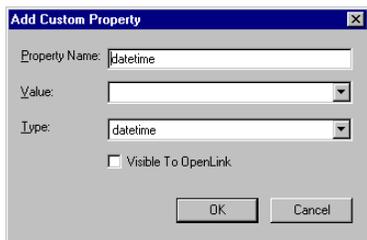
そのプランの「Properties」ダイアログ・ボックスが表示されます。



3. 「Custom Plan Properties」 タブで、「Add」 をクリックします。



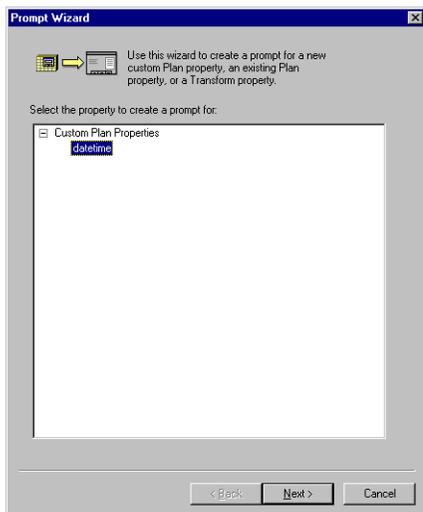
4. プロパティ名「**datetime**」を入力して、「**Type**」ドロップダウンから「**datetime**」を選択します。その後、「**OK**」をクリックします。



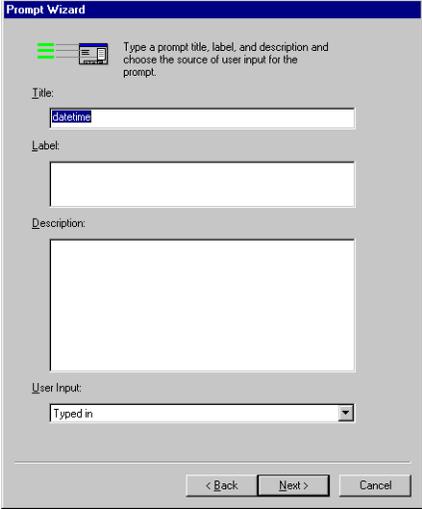
5. 「**Prompts**」タブで、「**New Prompt**」をクリックします。



6. 「**datetime**」プロパティを選択します。その後、「**Next**」をクリックします。



7. 「Title」フィールドに「datetime」と入力します。必要に応じて、datetimeのプラン・プロンプトの「Label」および「Description」を入力します。その後、「Next」をクリックします。



Prompt Wizard

Type a prompt title, label, and description and choose the source of user input for the prompt.

Title:

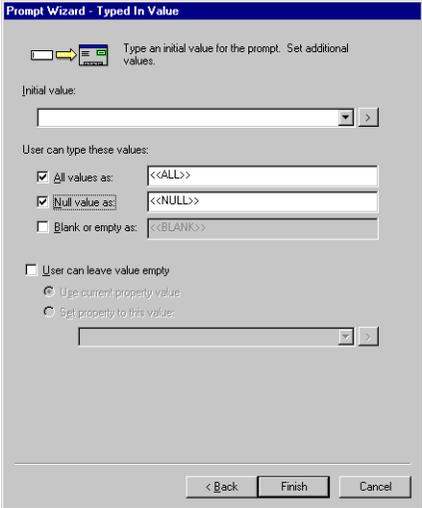
Label:

Description:

User Input:

< Back Next > Cancel

8. 「All values as」および「Null values as」をクリックします。その後、「Finish」をクリックします。



Prompt Wizard - Typed In Value

Type an initial value for the prompt. Set additional values.

Initial value:

User can type these values:

All values as:

Null value as:

Blank or empty as:

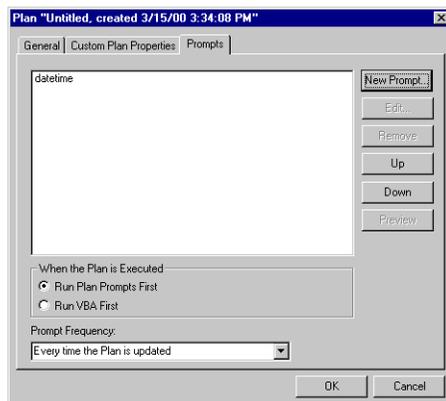
User can leave value empty

Use current property value

Set property to this value:

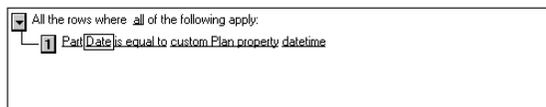
< Back Finish Cancel

9. 「OK」 をクリックします。

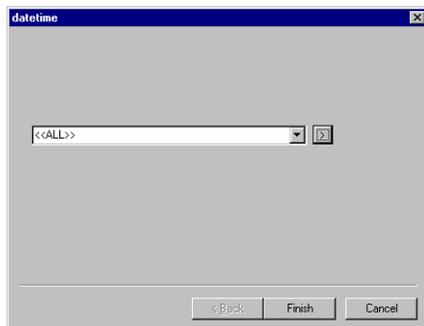


10. datetime のカスタム・プラン・プロパティ・フィルタをプランに追加します。

カスタム・プラン・プロパティ・フィルタの追加方法の詳細は、[116 ページ](#)の「フィルタでのカスタム・プラン・プロパティの使用」を参照してください。



11. 「Update」 をクリックして、プランを実行します。
datetime のプラン・プロンプトが表示されます。



12.  ボタンをクリックします。
13. プランに適用する日付を選択します。その後、「OK」をクリックします。



14. 「Finish」を選択します。
プランが更新されます。

StarMarts の設計

ここで説明する内容は次のとおりです。

- [概要](#)
- [演習 : StarMart の構築](#)
- [StarView の変更](#)
- [集計の環境のメンテナンス](#)

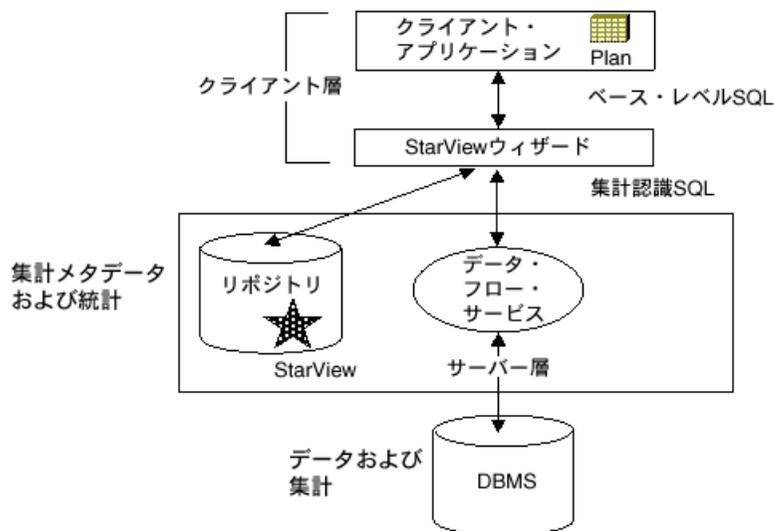
概要

集計とは、他のレコードを集計するファクト表のレコードです。集計は、平均や合計などの計算式を一連のレコードに適用して計算されます。StarMart を構築する背景には、ユーザーが一般的に求める集計のタイプを予期し、このデータがリクエストされる前に集計および格納するという方針があります。

集計では、プランのパフォーマンスを動的に向上させ、ユーザーからユーザーへ一貫したサマリーを提供できます。たとえば集計が使用可能な場合、WebLink ユーザーは、プランの更新時に高速かつ一貫性のある集計データを取得できます。プランが実行可能なすべての環境において、エンド・ユーザーが意識することなくパフォーマンス上の利点が発揮されます。

StarMart の構築には詳細な計画が不可欠です。特に、ユーザーが共通してリクエストする集計データの種類を特定する必要があります。この情報を使用して、ユーザーのニーズに合う集計スキーマを設計できます。集計によって、パフォーマンスを大きく向上させることができますが、明確な利点はそれぞれの状況で分析する必要があります。一般に、サマリーのレベルが高くなるにつれ、集計はより有益になります。

StarMart の主要な利点の 1 つは、集計ナビゲーションです。集計ナビゲーションは、ベース・レベル SQL を集計認識 SQL に変換します。このプロセスは、クライアント・ユーザーが集計表の存在を意識する必要がないことを意味します。次の図では、データ・マートの集計をナビゲートするプロセスを説明します。



クライアント層には、集計をナビゲートしメンテナンス・タスクを自動化するツールである **StarView** ウィザードがあります。集計表に関する情報は、**StarView** のリポジトリに格納されます。**MetaView** にアクセスすると、この情報は、**StarView** ウィザードがその情報にアクセスできるクライアント・マシン上にキャッシュされます。ユーザーがプランをワークスペースにドラックすると、**StarView** ウィザードは **SQL** を変換します。可能な場合、ベース・レベル表の名前は集計表によって置換されます。**Data Flow Service** は、この集計認識 **SQL** を受信し、必要な他の操作を実行し、実行するため **SQL** 文をデータベースに渡します。

集計メタデータがサーバー層に格納されるため、クライアント・アプリケーションを変更せずに集計表を削除および追加できます。

次の項では、**Design Studio** で集計を作成および管理するプロセスについて説明します。この章では、ユーザーがファクト、ディメンション、属性、メジャーなどの特定のディメンション用語に精通していることを前提としています。これらの用語の定義については、オンライン・ヘルプの用語集を参照してください。

演習 : StarMart の構築

このチュートリアルでは、サンプルの **StarMart** の構築手順について説明します。この演習では、**Oracle BAM Enterprise Link** のサンプルに含まれる **samples_gourmet_foods** データベースを使用します。

この演習では、始めに **BaseView** を作成し、次に **StarView** ウィザードを使用して、**StarView** を自動的に作成します。2つのディメンションの属性階層を編集して、**StarView** のすべてのメタデータが完全で正しいことを確認します。その後、**StarView** に定義されている集計ファクト表およびディメンション表を作成し、プランを実行してそれらを移入します。

ここで説明する内容は次のとおりです。

- [StarView エディタの表示](#)
- [StarView の作成](#)

StarView エディタの表示

StarView エディタを使用して、**StarViews** を作成、変更および削除します。また、**StarView** エディタでは、集計表を作成および移入し、表の使用方法に関する統計を取得します。

StarView エディタは、**Design Studio** の **BaseView** エディタに組み込まれています。**BaseView** エディタへのアクセス方法および表リストの表示方法の詳細は、第4章を参照してください。

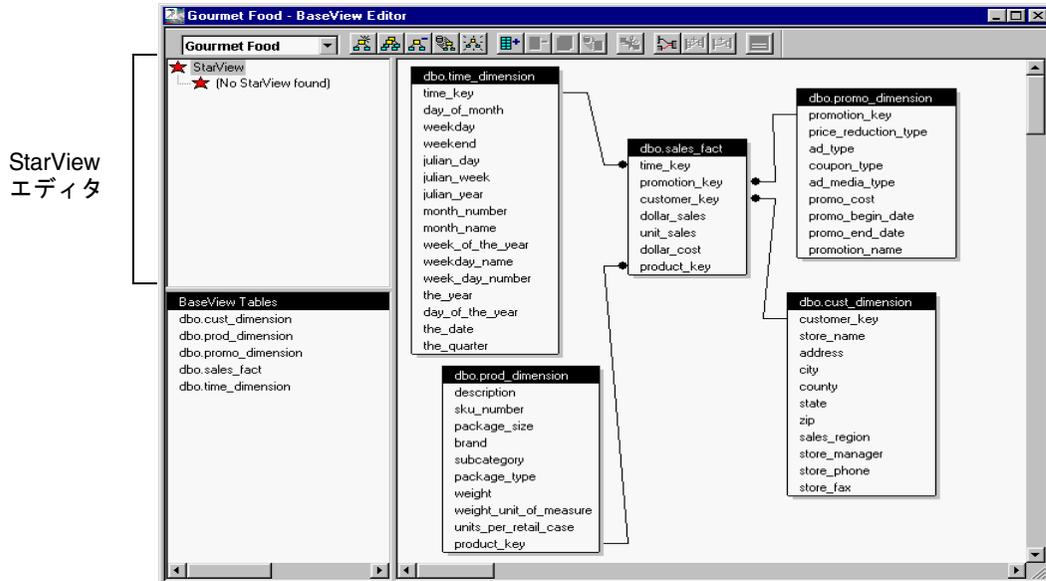
StarView エディタを表示するには、次の手順を実行します。

1. BaseView 表リストで、BaseView エディタ・ウィンドウの上部の縁のハンドルにカーソルを置きます。

カーソルは、水平に分かれた矢印になります。



2. ウィンドウの下部にカーソルをドラックして、StarView エディタを表示します。



StarView の作成

StarView は、BaseView と MetaView の中間層です。StarViews によって、StarViews ウィザードにプランで使用する集計表およびそれらの作成方法が示されます。StarViews の情報には、集計表および使用状況の統計の説明、メジャーおよび集計関数、属性および属性階層が含まれます。

ここで説明する内容は次のとおりです。

- BaseView の作成
- StarView の作成
- 属性階層の編集
- 集計表の作成
- 集計表の移入

既存の StarView を変更するには、144 ページの「StarView の変更」を参照してください。

BaseView の作成

最初に、サンプルの StarView に使用する BaseView を作成します。

BaseView を作成するには、次の手順を実行します。

1. Design Studio で、「Tools」 → 「BaseView Editor」を選択します。

2. ツールバーの「New BaseView」ボタンをクリックします。

「Define BaseView」ダイアログ・ボックスが表示されます。

3. 「Define BaseView」ダイアログ・ボックスに次の情報を入力します。

BaseView Name: BaseView の名前に、**StarFoods** と入力します。

Login Name: samples_gourmet_food データベースのログイン名に、sa と入力します。

Password: このフィールドは空白のままにします。

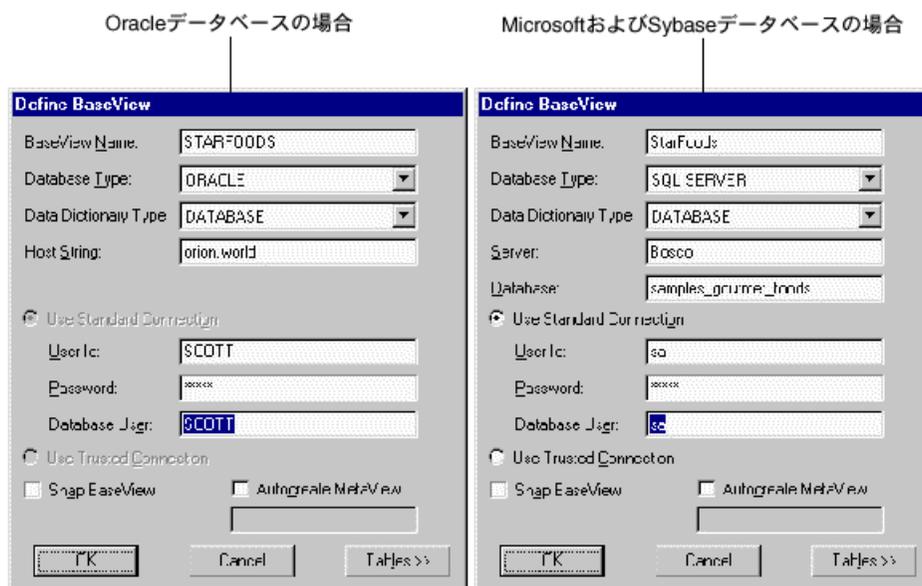
Server: サンプル・データベースが配置されているサーバーの名前を入力します。

Database: BaseView フォームを作成するデータベースの名前に、**samples_gourmet_food** と入力します。

Database User: 「Login Name」のデータベース・ユーザー・アカウントが「Login Name」と同じ場合、このフィールドには何も入力しません。

Database Type: 「SQL Server」を選択します。

Database Dictionary Type: 「Database」を選択します。



StarView ウィザードでは、StarMart に集計表を作成する際にこの接続情報を使用します。BaseViews の作成の詳細は、第 3 章を参照してください。

4. 「OK」をクリックして、BaseView を作成します。
5. BaseView の作成後、スター形式の表を sales_fact 表とともに中央に配置します。
6. 次の結合を BaseView に追加します。
 - time_dimension 表の time_key から sales_fact 表の time_key へのドラッグ
 - cust_dimension 表の customer_key から sales_fact 表の customer_key へのドラッグ
 - prod_dimension 表の product_key から sales_fact 表の product_key へのドラッグ
 - promo_dimension 表の promotion_key から sales_fact 表の promotion_key へのドラッグ

StarView の作成

次に、BaseView の StarView を作成します。BaseView と同様、StarView はエンド・ユーザーには意識されません。

1. StarView エディタを右クリックして、「New StarView」を選択します。
「StarView Wizard」ダイアログ・ボックスが表示されます。



2. 「Yes」をクリックして、StarView を自動的に作成します。

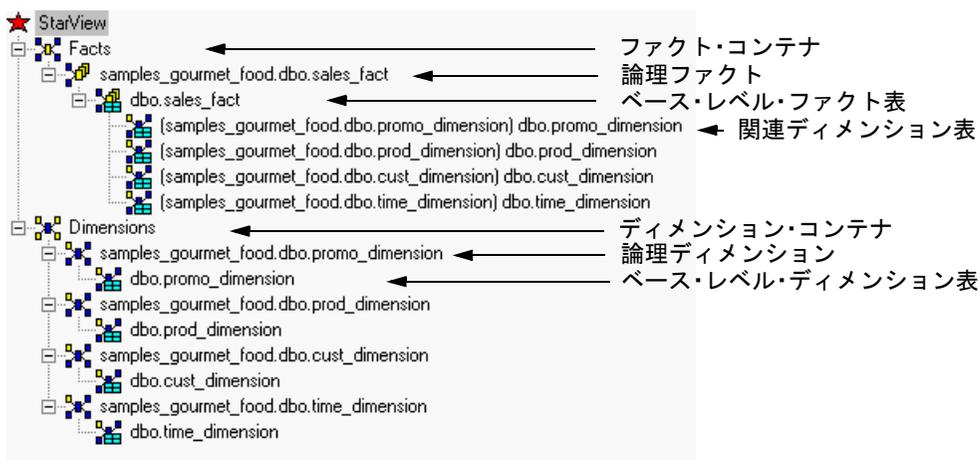
StarView ウィザードでは、StarFoodsBaseView の結合のパターンおよびカーディナリティを使用して、ファクト表およびディメンション表を特定します。スター・スキーマには特徴的な形があり、これによってディメンション表の各レコードがファクト表の多くのレコードを示すことができます。表がこのパターンに構造化される場合、StarView ウィザードでは中央の表をファクト表として、他の表をディメンション表として特定します。

StarView ウィザードで StarView を作成してから、この解釈をそのまま使用するか、または変更するかを決めることができます。簡単なスターで構造化された StarFoods と同様のベース・レベル・スキーマの場合、必要な変更はわずかです。変更されたスター・スキーマの場合、StarView ウィザードでは StarView の一部を作成し、特定できない表を省略します。

注意 :StarView ウィザードでは既存のサマリー表を認識しません。既存のサマリー表に対して新しい StarView を作成する場合には、160 ページの「既存の集計表の使用」を参照してください。

3. 新しい StarView のすべてのレベルを展開してコンテンツを表示します。

図 1: StarView の図

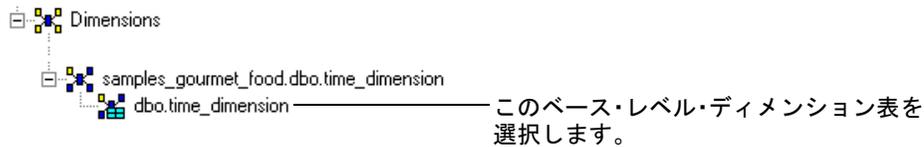


これらのアイテムは、StarView ウィザードに集計の作成方法および集計へのナビゲート方法を伝えます。ファクト・コンテナには、論理ファクト `sample_gourmet_food.dbo.sales_fact` が含まれます。ベース・レベル・ファクト表 `dbo.sales_fact` は、論理ファクトの下にネストされます。次のレベルには、ファクト表に依存したディメンション表のリストがあります。

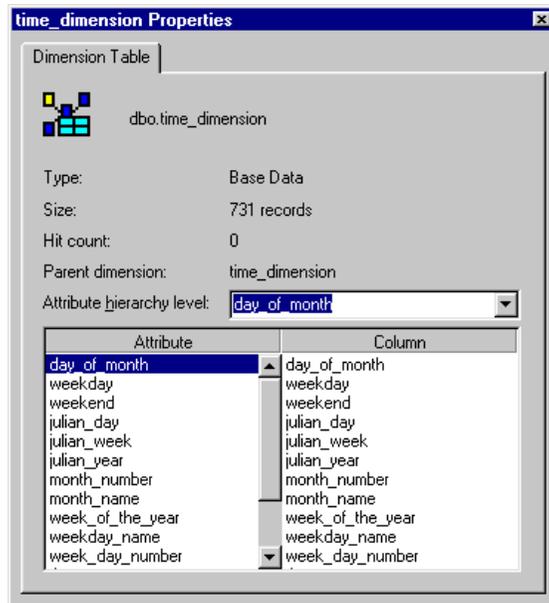
ディメンション・コンテナでは、BaseView で見られる論理ディメンションが示されます。関連したベース・レベル・ディメンション表が各論理ディメンションの下にネストされます。

StarView ウィザードで StarView を作成してから、集計表およびディメンションが正しく特定され、ベース・レベル表にマップされていることを確認する必要があります。また、列マッピングが完了しており、正しいことも確認します。

- StarView のベース・レベル表 `dbo.time_dimension` を右クリックして、「Properties」を選択します。



「Dimension Table Properties」ダイアログ・ボックスが表示されます。



ベース・レベル時間ディメンション表のすべての列がダイアログ・ボックスの右側に示され、属性が各列にマップされています。デフォルトでは、属性名は列名と同じです。属性を削除し、新しい名前を追加して、属性の名前を変更できます。手順については、[144 ページの「StarView の変更」](#)を参照してください。

- StarView の残りのすべてのファクト表およびディメンション表に対して手順 4 を繰り返します。

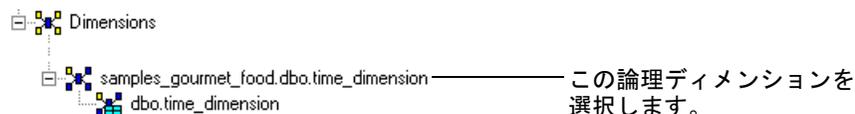
すべてのベース・レベルのファクト表およびディメンション表のプロパティを確認すると、属性階層を編集し集計関数を割り当てることができるようになります。

属性階層の編集

属性とは、ディメンション表の列を表現する論理エンティティです。StarView では、各論理ディメンションの階層に、関連する属性を配置し、これによって集計します。この階層は、ディメンションのベース・レベル・ディメンション表および集計ディメンション表に適用されます。

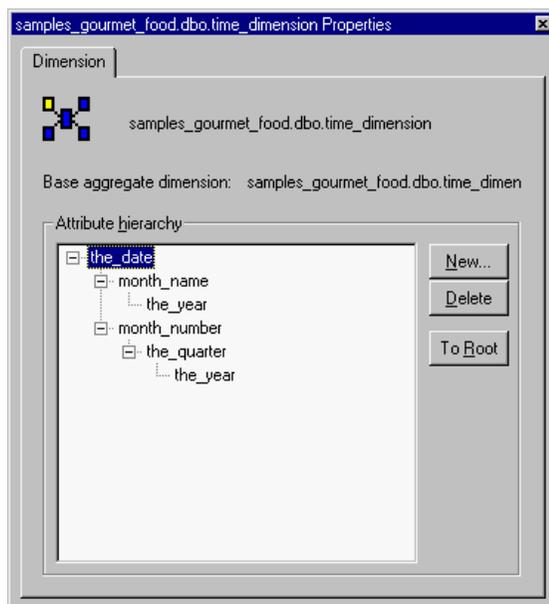
この StarView の場合、時間ディメンションおよび製品ディメンションの属性階層を編集する必要があります。販売促進ディメンションおよび顧客ディメンションについては、これらのディメンションに集計表が必要ないため、この手順を省略できます。

1. StarView の `samples_gourmet_food.dbo.time_dimension` を右クリックして、「**Properties**」を選択します。ディメンション表ではなく、論理ディメンションをクリックします。



デフォルトでは、StarView ウィザードは属性階層のルート・レベルにディメンション表のキー以外の最初の列を配置します。

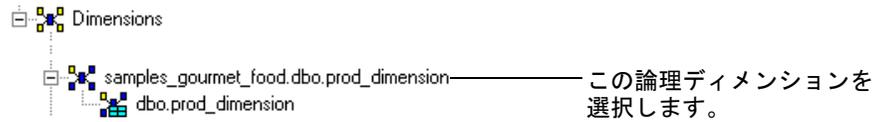
2. 時間ディメンションを次のマルチパス階層に編成します。属性階層の編集については、[154 ページの「属性階層の編集」](#)を参照してください。



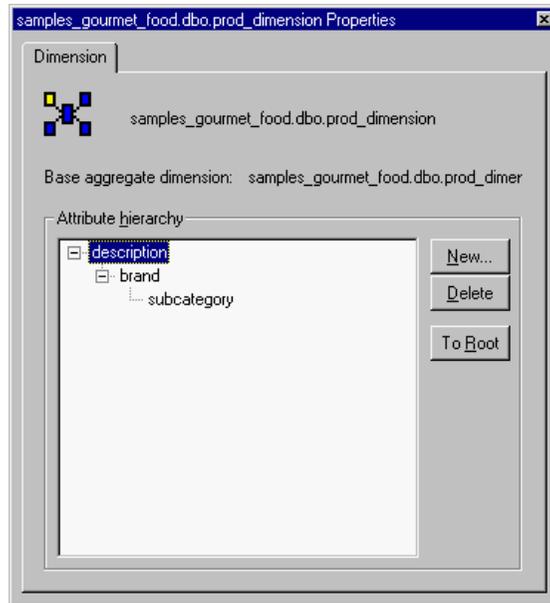
この階層は、ディメンション全体に当てはまります。このディメンションのすべての表、ベース・レベルまたは集計には、最も詳細なタイプのデータとして、これらのいずれかの属性があります。この属性は、ディメンション表のレベルと呼ばれています。

上に示す階層では、`the_date` は、最も詳細なデータを含むディメンション列です。この属性は、1つのドリル・パスで、月から年にロール・アップし、もう1つのドリル・パスで、月、四半期、年にロール・アップします。

3. StarView の `samples_gourmet_food.dbo.prod_dimension` をクリックします。ディメンション表ではなく論理ディメンションを選択します。



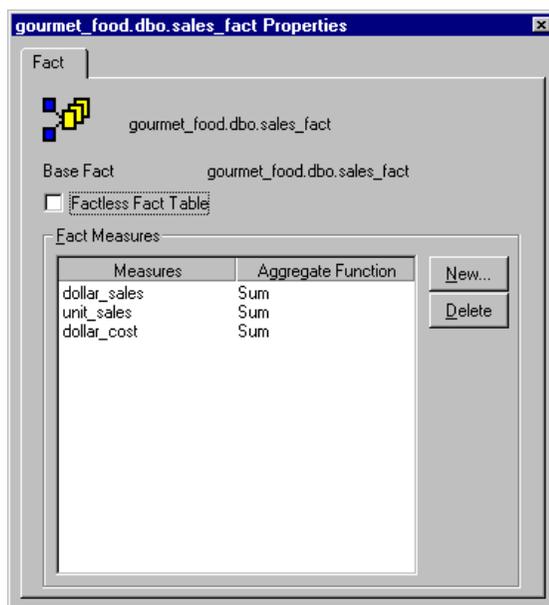
4. 右クリックして「**Properties**」を選択し、属性階層を表示します。
5. 製品ディメンションを次の階層に編成します。



この階層では、説明列に最も詳細なデータが含まれます。この属性は、銘柄、さらにサブカテゴリにロール・アップします。

6. 「Dimension Properties」ダイアログ・ボックスを閉じます。

7. 論理ファクト `sample_gourmet_food.dbo.sales_fact` を右クリックして「Properties」を選択し、「Fact Properties」ダイアログ・ボックスを表示します。



ベース・レベル・ファクト表のすべての列が、このダイアログ・ボックスの「Measures」の下に示されます。これらの列は、論理ファクトの基本です。また、すべての関連する集計ファクト表の列は、この論理ファクトに属します。

このサンプルの StarView の場合、各メジャーに割り当てられるデフォルトの Sum 集計関数を使用します。これらのメジャーが加算ではない場合、Count などの異なる関数を使用する場合があります。

8. 「Fact Properties」ダイアログ・ボックスを閉じます。

集計表の作成

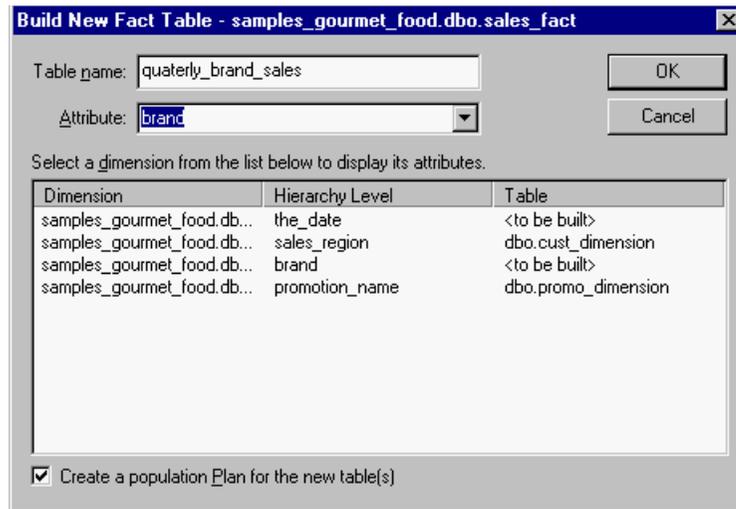
StarView が完了したら、次の手順では集計表を作成します。集計表を作成する際に、1つ以上の集計ファクト表と1つ以上の集計ディメンション表を作成します。この例では、1つの集計ファクト表と2つの関連したディメンション表を作成します。

集計表に集計できるデータの単位は、ベース・レベル・ファクト表およびベース・レベル・ディメンション表の単位によって異なります。ベース・レベル・データに存在する集計表の詳細の集計のみ実行できます。

集計表を作成するには、次の手順を実行します。

1. StarView の「**Facts**」 コンテナをダブルクリックして、論理ファクトを表示します。
2. `samples_gourmet_food.dbo.sales_fact` を右クリックして、「**Build New Fact Table**」を選択します。
「Build New Fact Table」ダイアログ・ボックスが表示されます。
3. 「Table name」フィールドに `quarterly_brand_sales` と入力します。
4. 「Dimension」リストの `sample_gourmet_food.dbo.prod_dimension` をクリックして、「Attribute」ドロップダウン・リストから「**brand**」を選択します。
5. 「Dimension」リストの `sample_gourmet_food.dbo.time_dimension` をクリックして、「Attribute」ドロップダウン・リストから「**the_quarter**」を選択します。

「Build New Fact Table」ダイアログ・ボックスには次の設定が含まれます。

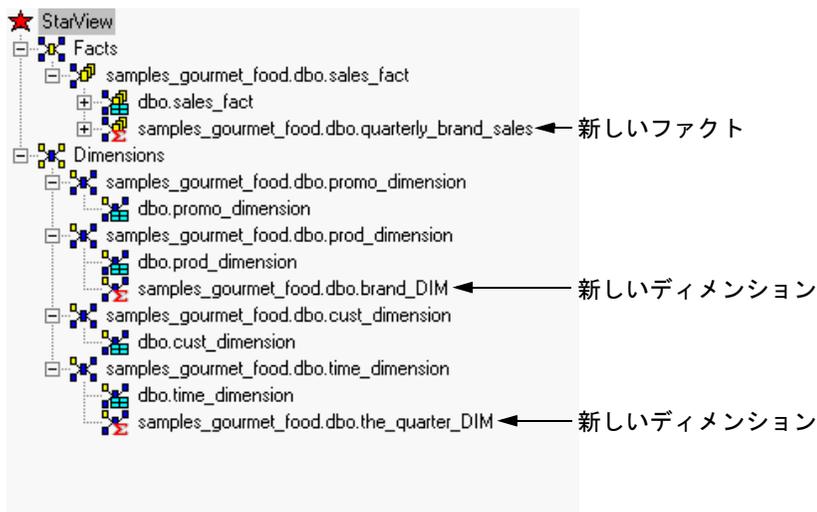


販売促進ディメンションおよび顧客ディメンションの場合、ベース・レベル属性が使用されます。これらのディメンションには、集計ディメンション表は必要ありません。

6. 「**OK**」をクリックします。

StarView ウィザードは、この BaseView のデータベースに新しい集計ファクト表および新しい集計ディメンション表を作成し、集計表を移入するためにプランを作成します。「Build New Fact Table」ダイアログ・ボックスに示されるすべてのディメンションに対して集計ディメンション表が存在しない場合、ディメンション表が作成されます。

新しい論理ファクトおよび依存アイテムが StarView に追加され、新しい集計表が Base View エディタ・ウィンドウに表示されます。



2つの新しい集計ディメンション表 `the_quarter_DIM` および `brand_DIM` には、集計ディメンション・レコードが含まれます。StarView ウィザードでは、各ディメンションの属性レベルを使用して、集計表に名前を付けます。集計ファクト表 `quarterly_brand_sales` には、`brand` および `quarter` によって集計されたファクトが含まれます。

これらの表の `Part` および `Category` は、この BaseView のどの MetaView にも追加されません。StarView ウィザードでは、パフォーマンスに有効な場合に、これらの表からデータを自動的にリクエストします。

集計ファクト表は、合成キーによって、集計されたファクトを示すディメンション表に結合されます。新しい集計表が必要なかったディメンション `promo_dimension` および `cust_dimension` は、ベース・レベル・ファクト表 `sales_fact`、および集計ファクト表 `quarterly_brand_sales` の両方に結合されます。

集計表の移入

StarView を作成する際に、集計の環境を設計します。集計表が作成されたら、次に集計データを集計表に移入します。この例で、StarView ウィザードでは、集計移入プランを作成しました。プランは、ベース・レベル表から選択し、集計関数を適用し、新しい集計表に挿入する SQL を実行します。通常は、後でプランを実行するようにスケジュールする必要があります。このサンプル・データベースにはほとんどレコードがないため、対話方式で実行できます。

サンプル移入プランは、次の手順で実行します。

1. ワークスペースにハンドルをドラックして、データ・フロー・エディタを表示します。
2. Plan Bin から、移入プラン `quarterly_brand_sales` をワークスペースにドラックします。

3. データ・フロー・エディタで、「**Query Step**」をダブルクリックして、SQL テキストを表示します。
SQL エディタに、StarView ウィザードで作成された集計認識 SQL が表示されます。
4. SQL エディタを閉じ、「**Update**」をクリックしてプランを実行します。
プランが実行され、サンプル集計表が移入されます。
5. StarView エディタで「**StarView**」をクリックします。
6. 右クリックして、「**Refresh Statistics**」を選択します。

これで、ユーザーが銘柄、四半期または年によってファクト・レコードをリクエストするプランを実行すると、新しい集計ファクト表の事前集計されたレコードが使用されるようになります。StarView ウィザードでは、ユーザーが意識しなくても、データへの最短パスが使用されます。

StarView の変更

StarView を変更する際に、考慮すべき 2 つの層があります。それは、StarView のメタデータ層とソース・データベースの物理層です。これらの層は、互いに独立しています。データベースで集計表を変更しなくても、StarView を変更することができます。ただし StarView に対する変更は、集計表のデータがリフレッシュされるまで、集計には反映されません。

StarView は、StarView ウィザードを使用しなくても、手動で StarView を編集することによって作成できます。

ここで説明する内容は次のとおりです。

- [StarView の編集](#)
- [StarView の削除](#)

StarView の編集

StarView は、集計ファクトとディメンション表の関係を定義する構造化されたリストです。137 ページの「[StarView の図](#)」は StarView の構造を示します。

注意： 集計表がすでに作成されている場合に StarView を編集すると、集計表が使用不能であると StarView で表示される場合があります。表に影響する変更を行うと、警告ダイアログ・ボックスが表示されます。

StarView を編集するには、次のようにします。

- 既存のアイテムを削除して、それを正しい場所に追加し、必要なプロパティを指定します。

注意： BaseView から表を削除すると、StarView にある、表に依存するすべての表が StarView から削除されます。

StarView のアイテムが選択される際に、右クリックして「**Properties**」を選択してプロパティ・シートを表示します。

ここで説明する内容は次のとおりです。

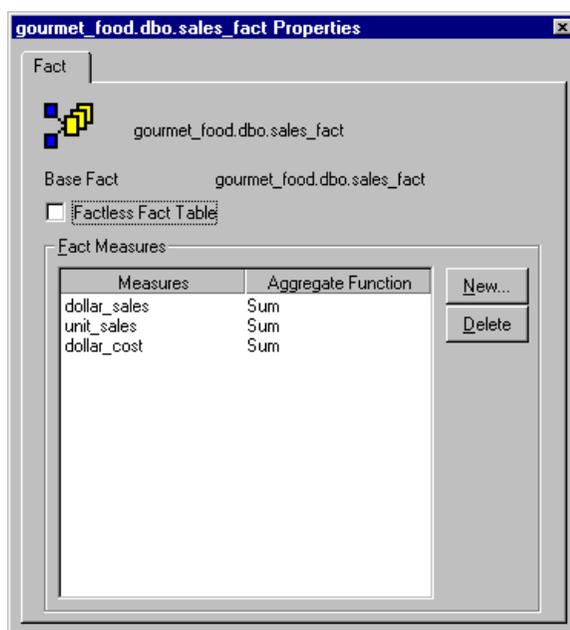
- ファクト・プロパティの変更
- ファクトレス・ファクト表の使用
- ディメンション・プロパティの変更

ファクト・プロパティの変更

StarView のファクトのプロパティを変更するには、次の手順を実行します。

- ファクト・コンテナの下のファクト表を右クリックし、「**Properties**」を選択します。

「Fact Properties」ダイアログ・ボックスが表示されます。



StarView ウィザードは、このファクトのすべてのメジャーをベース・レベル・ファクト表に自動的に関連付けます。この表の名前は、ベース・ファクト集計フィールドに表示されます。

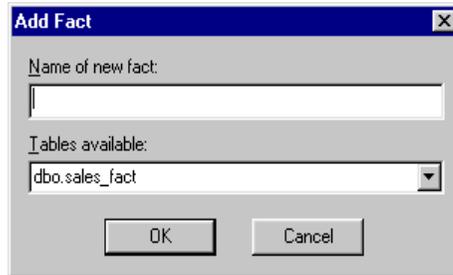
このファクトのメジャーは、ダイアログの左側に表示されます。現在、各メジャーに割り当てられている集計関数は、右側に表示されます。

StarView へのファクトの追加

ファクトは、ベース・レベル・ファクト表にマップできる論理エンティティです。ファクトには、任意の名前を指定できます。通常、ファクトには、ファクト表、メジャーと列のマッピング、集計関数、およびすべての関連する集計ファクト表が含まれます。論理ファクトを追加して、これらの関連する要素をグループ化します。また、最初にファクトを削除して名前を変更した後に、ファクトを追加する場合があります。

StarView に論理ファクトを追加するには、次の手順を実行します。

1. 「StarView」で右クリックして、「Add Fact」を選択します。
「Add Fact」ダイアログ・ボックスが表示されます。



2. ファクトの名前を入力します。この名前は、StarView 内で一意である必要がありますが、他の制限はありません。名前は、データベース表または列の名前と一致させる必要はありません。
3. BaseView のベース・レベル・ファクト表の名前を選択して、ドロップダウン・リストからこのファクトと関連付けます。
4. 「OK」をクリックします。

新しいファクトは、StarView の下の階層に追加されます。

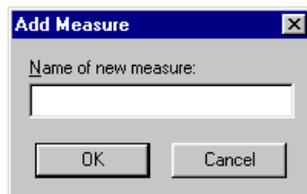
ファクトへのメジャーの追加

メジャーとは、ベース・レベル・ファクト表の列にマップできる論理エンティティです。メジャーには、任意の名前を指定できます。集計表を作成すると、集計ファクト表の列には、ベース・レベル・ファクト列ではなく、メジャーに基づいて名前が付けられます。

最初にメジャーを削除して名前を変更した後に、メジャーを追加する場合があります。ベース・レベル・ファクト列に論理的に同等なものがない場合、メジャーを追加する場合があります。

新しいメジャーを現行のファクトに追加するには、次の手順を実行します。

1. 「Fact Properties」ダイアログ・ボックスで、「New」ボタンをクリックします。
「Add Measure」ダイアログ・ボックスが表示されます。



- 新しいメジャーの名前を入力します。この名前は、StarView 内で一意である必要がありますが、他の制限はありません。名前は、データベース表または列の名前と一致させる必要はありません。
- 「OK」をクリックします。

ファクトは、「Fact Properties」ダイアログ・ボックスに追加され、Sum 集計関数に割り当てられます。「Fact Table properties」ダイアログ・ボックスでは、このメジャーはどの列にもマップされません。集計ファクト表で使用するには、このメジャーをデータベース列と関連付ける必要があります。

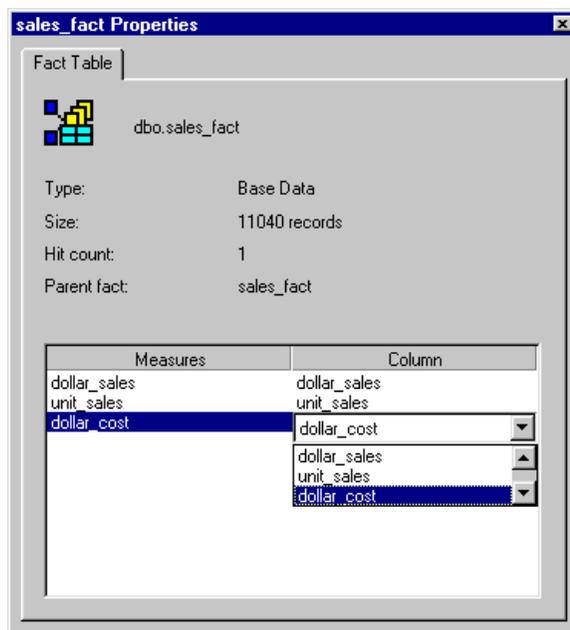
ファクト表の列とメジャーの関連付け

メジャーは、ベース・レベル・ファクト表の列にマップされた場合のみ意味があります。列またはメジャーのいずれかを追加した場合、属性をディメンション表の列にマップする必要がある場合があります。

メジャーをファクト表の列と関連付けるには、次の手順を実行します。

- StarView のファクト表の名前を右クリックして、「Properties」を選択します。

「Fact Table Properties」ダイアログ・ボックスが表示されます。



- 「Fact Table Properties」ダイアログ・ボックスで、変更するメジャーに対応する列の値をクリックします。
列の値がドロップダウン・リストになります。
- 新しいファクト表の列を選択して、リストからこのメジャーと関連付けます。
- ダイアログ・ボックスを閉じて、変更を保存します。

変更は、StarView にすぐに適用されます。これで、この列を含む集計ファクト表を作成できました。

集計関数の割当て

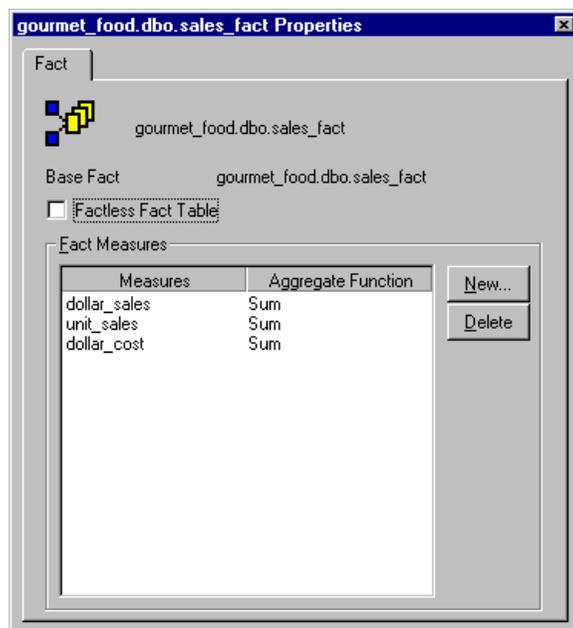
集計関数は、ベース・レベル・レコードに適用される数式です。結果は、集計ファクト表の集計レコードです。Oracle BAM Enterprise Link では、次のような一般的な SQL 集計関数をサポートします。

集計関数	説明	例
SUM	一連のレコードの合計を計算します。	20+20+30+40=110
MIN	最小値を算出します。	20
MAX	最大値を算出します。	40
COUNT	問合せによって返されたレコード数を計算します。	4
AVG	平均値を計算します。 (SUM/COUNT)	110/4= 27.5

指定する関数は、集計移入プランが作成される際、SQL 文に含まれます。デフォルトでは、メジャーは SUM 集計関数に割り当てられます。集計関数は、前述の表に示されているどの関数にでも変更できます。

集計関数を変更するには、次の手順を実行します。

1. StarView の論理ファクトの名前をクリックします。
2. 右クリックして、「**Properties**」を選択します。
「Fact Properties」ダイアログ・ボックスが表示されます。



- 変更するメジャーに対応する「Aggregate Function」の値をクリックします。

「Aggregate Function」の値がドロップダウン・リストになります。

- リストから新しい集計関数を選択します。
- ダイアログ・ボックスを閉じて、変更を保存します。

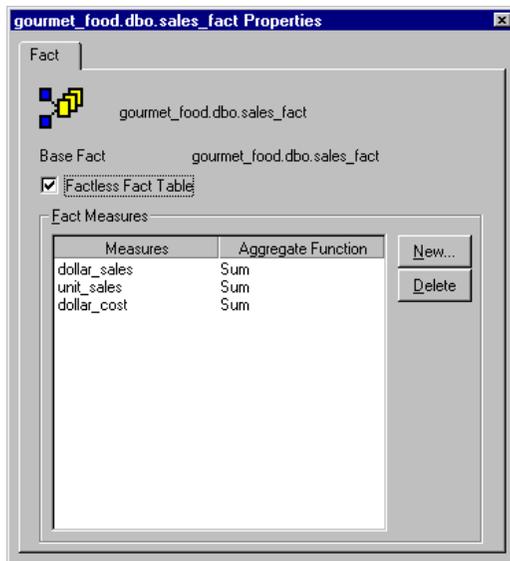
変更は、StarView にすぐに適用されます。次回、集計表にデータを移入すると、新しい集計関数がこのメジャーの計算に使用されます。

ファクトレス・ファクト表の使用

ファクトレス・ファクト表は、ディメンションが相互にどのように関連するかを記録する関係表です。ファクトレス・ファクト表は、数値メジャーを含む場合も、含まない場合もあります。ファクトレス・ファクト表で定義する情報のタイプは、通常はキー値の有無です。たとえば、施設使用率の測定では、数量ではなくイベントを比較します。

ファクトレス・ファクト表を使用すると、StarView を使用する BaseView にアクセスするプランに、関連するメジャーがない複数のディメンションからの Part を含めることができます。ユーザーは、Part を異なるディメンションからワークスペースにドラックできます。ファクトレス・ファクト表が存在し、特定のディメンションを関係付けている場合、そのファクトレス・ファクト表は問合せに使用されます。

ファクトレス・ファクト表として、「Logical Fact」プロパティ・シートに表を指定します。



「Factless Fact Table」チェック・ボックスが選択されている場合、表は関係のみに使用されます。表にメジャーも含まれている場合、プランではメジャーを取得できません。

ファクトレス・ファクト表に基づいて集計ファクト表を作成することもできます。集計ファクト表のキーは、高い属性レベルでディメンション関係を集計します。

注意： ファクトレス・ファクト表に、通常のファクトおよびファクトレス・ファクトを組み合わせて含めることはできません。また、複数のディメンション表にドリルすることはできません。

ここで説明する内容は次のとおりです。

- [ファクトからのメジャーの削除](#)
- [StarView からのファクトの削除](#)

ファクトからのメジャーの削除

論理ファクトからメジャーを削除すると、メジャー名とファクト表の列のマッピングが削除されます。これは、ベース・レベル・ファクト表の関連する列から値を集計できないことを意味します。名前の変更、または集計ファクト表にこの列を含めないことを目的として、メジャーを削除する場合があります。

論理ファクトからメジャーを削除するには、次の手順を実行します。

1. 「Fact Properties」ダイアログ・ボックスで、メジャーの名前をクリックします。
2. 「Delete」ボタンをクリックします。

ファクトからメジャーが削除されます。メジャーは論理エンティティであるため、関連するベース・レベル・ファクト表の列は影響されません。

StarView からのファクトの削除

StarView から論理ファクトを削除すると、メジャーと列のマッピングおよび集計関数を含むファクトの定義が削除されます。また、StarView からファクト表などのすべての依存アイテムも削除されます。BaseView、または基礎となるデータベースには影響しません。

StarView から論理ファクトを削除するには、次の手順を実行します。

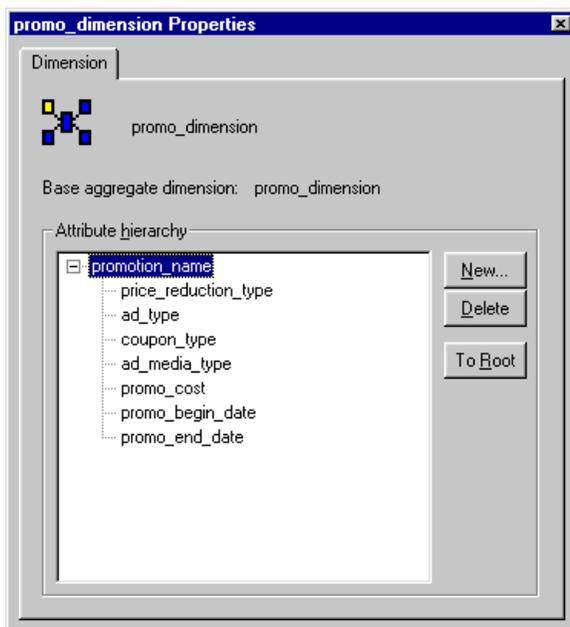
1. StarView のファクトを右クリックし、「Delete」を選択します。
プロンプトによって、ファクトおよびその関連ファクト表を削除することを確認するように求められます。
2. 「OK」をクリックします。

ディメンション・プロパティの変更

StarView の論理ディメンション・プロパティを変更するには、次の手順を実行します。

1. StarView エディタのディメンションの名前をクリックします。
2. 右クリックして、「**Properties**」を選択します。

「Dimension Properties」ダイアログ・ボックスが表示されます。このダイアログ・ボックスを使用して、ディメンションの属性階層を編集します。属性階層の編集方法については、[154 ページの「属性階層の編集」](#)を参照してください。



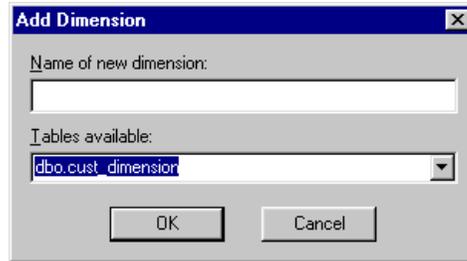
StarView へのディメンションの追加

ディメンションは、ベース・レベル・ディメンション表にマップできる論理エンティティです。ディメンションには、任意の名前を指定できます。通常、ディメンションには、ディメンション表、属性と列のマッピング、集計階層、およびすべての関連する集計ディメンション表が含まれます。ディメンションを追加して、これらの関連する要素をグループ化します。また、最初にディメンションを削除して名前を変更した後に、ディメンションを追加する場合があります。

StarView に論理ディメンションを追加するには、次の手順を実行します。

1. ディメンション・コンテナを右クリックし、「Add Dimension」を選択します。

「Add Dimension」ダイアログ・ボックスが表示されます。



2. ディメンションの名前を入力します。この名前は、この StarView 内で一意である必要がありますが、他の制限はありません。名前は、データベース表または列の名前と一致させる必要はありません。
3. BaseView のディメンション表の名前を選択して、ドロップダウン・リストからこのディメンションと関連付けます。
4. 「OK」をクリックします。

新しいディメンションが、ディメンションの下の StarView に追加されます。

ディメンションへの属性の追加

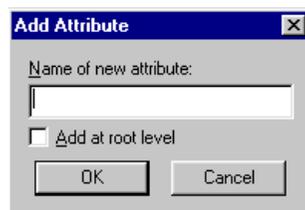
属性は、ベース・レベル・ディメンション表の列にマップできる論理エンティティです。属性には、任意の名前を指定できます。集計表を作成すると、集計ディメンション表の列には、ベース・レベル・ディメンション列ではなく、属性に基づいて名前が付けられます。

最初に属性を削除して名前を変更した後に属性を追加する場合があります。ベース・レベル・ディメンション列に論理的に同等なものがない場合、属性を追加する場合があります。

新しい属性をディメンションに追加するには、次の手順を実行します。

1. ディメンションのプロパティ・シートで、「New」ボタンをクリックします。

「Add Attribute」ダイアログ・ボックスが表示されます。



2. 新しい属性の名前を入力します。この名前は、StarView 内で一意である必要がありますが、他の制限はありません。名前は、データベース表または列の名前と一致させる必要はありません。
3. この属性を属性階層の最上位にするには、「Add at root level」をクリックします。表の中で最も詳細な属性は、ルート・レベルにあります。
4. 「OK」をクリックします。

新しい属性が、「Dimension Properties」ダイアログ・ボックスに追加されます。集計ディメンション表の属性を使用するには、それをデータベース列にマップする必要があります。

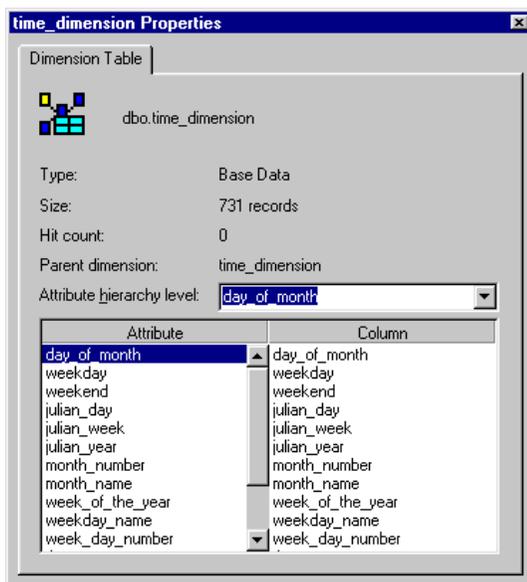
属性のディメンション表の列との関連付け

属性は、ベース・レベル・ディメンション表の列にマップされる場合のみ意味があります。列または属性のいずれかを追加した場合、属性をディメンション表の列にマップする必要があります。

属性をディメンション表の列にマップするには、次の手順を実行します。

1. 属性階層のディメンション表の名前を右クリックして、「Properties」を選択します。

「Dimension Table Properties」ダイアログ・ボックスが表示されます。



注意: 「Attribute hierarchy level」の値は変更しないでください。このリストは、トラブルシューティング専用です。ディメンションの階層レベルを設定するには、最初に「Dimension Properties」ダイアログ・ボックスの階層を編集して、「Build New Fact Table」ダイアログ・ボックスで集計のレベルを選択します。手順については、[141 ページの「集計表の作成」](#)を参照してください。

2. 「Dimension Table Properties」ダイアログ・ボックスで、変更する属性に対応する列の値をクリックします。

列の値がドロップダウン・リストになります。

3. 新しいディメンション表の列を選択して、リストからこの属性と関連付けます。
4. ダイアログ・ボックスを閉じて、変更を保存します。

変更は、StarView にすぐに適用されます。これで、この列を含む集計ディメンション表を作成できました。

属性階層の編集

属性階層では、ディメンションの属性間の関係を定義します。これらの関係は、集計の作成方法に影響します。

階層にはベース・レベル・ディメンション表の列名が含まれ、階層と集計表は独立しています。最上位のアイテムは、最も詳細なデータを含むディメンション列です。ネストされた各アイテムは、その上のアイテムよりも大まかなレベルです。

注意： 集計ディメンション表がすでに作成されている場合に属性階層を編集すると、その表が使用不能であると表示される場合があります。ディメンション表を無効にする可能性がある変更を行なうと、この表を削除、再作成および再移入するように警告ダイアログ・ボックスが表示されます。

属性階層を編集するには、次の手順を実行します。

1. その集計の階層で論理ディメンションの名前を右クリックし、「**Properties**」を選択します。
2. 「**Dimension Properties**」ダイアログ・ボックスで、属性階層の最上位のアイテムをダブルクリックして展開します。
3. 最上位のアイテムがディメンションの最も詳細なデータを含む列でない場合、最も詳細な列をクリックして、「**To Root**」をクリックします。
4. 属性をドラッグ・アンド・ドロップして、階層でそれらを再配置します。別の属性を記述または修飾する各属性は、その属性の下にネストされます。階層の属性を移動すると、下にネストされているすべての属性も移動されます。
5. 階層構造に適合しない属性を削除するには、階層の属性をクリックし、「**Delete**」をクリックします。

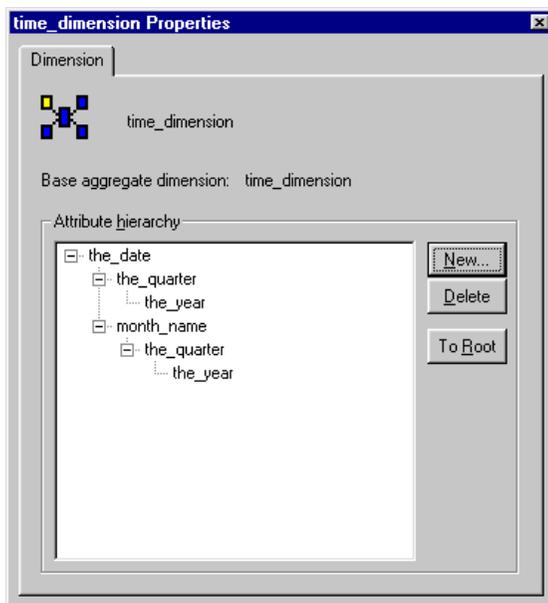
階層の属性を削除すると、ネストされたすべての属性も削除されます。

変更は、StarView にすぐに適用されます。

集計を作成すると、ネストされた属性は、集計ディメンション表に含まれます。たとえば、階層が日、月、四半期および年の属性を関連付け、月レベルで集計ディメンション表を作成する場合、ディメンション表には四半期および年の属性も含まれます。プランが四半期または年の情報をリクエストする場合、StarView ウィザードはその表にナビゲートします。

マルチパス階層の定義

マルチパス階層は、データへの複数のドリル・パスをサポートする属性階層です。たとえば、時間ディメンションは、四半期および年、また月、四半期および年に従って、論理的に編成できます。



このマルチパス階層は、詳細なデータを必要とするユーザーだけでなく、計画のために会計データが必要なユーザーに役立ちます。

マルチパス階層を定義するには、属性階層のアイテムをコピーして、それらを新しい場所にドラックします。階層の属性をコピーすると、ネストされたすべての属性もコピーされます。

マルチパス階層を定義するには、次の手順を実行します。

1. 「Dimension Properties」ダイアログ・ボックスで、コピーする階層の属性をクリックします。
2. [Ctrl] キーを押したまま、属性をドラックし、新しい場所にコピーを作成します。

コピーするすべての属性に対してこの手順を繰り返します。

ディメンションからの属性の削除

論理ディメンションから属性を削除すると、属性名とディメンション表の列のマッピングが削除されます。これは、集計ファクト表で、この属性では集計できないことを意味します。名前の変更、または集計ディメンション表にこの列を含めないことを目的として、属性を削除する場合があります。

現行のディメンションから属性を削除するには、次の手順を実行します。

1. 論理ディメンションのプロパティ・シートで、属性階層の属性の名前をクリックします。
2. 「Delete」ボタンをクリックします。

属性がディメンションから削除されます。すべてのネストされた属性も StarView から削除されます。属性は論理エンティティであるため、関連するベース・レベル・ディメンション表の列には影響ありません。

StarView からのディメンションの削除

StarView からディメンションを削除すると、論理ディメンションの定義が削除されます。これには、属性と列のマッピングおよび属性階層が含まれます。依存したディメンション表も StarView から削除されます。BaseView、または基礎となるデータベースには影響ありません。

StarView からディメンションを削除するには、次の手順を実行します。

1. StarView のディメンションを右クリックし、「Delete」を選択します。
プロンプトによって、ディメンションおよびその関連ディメンション表を StarView から削除することを確認するように求められます。
2. 「OK」をクリックします。

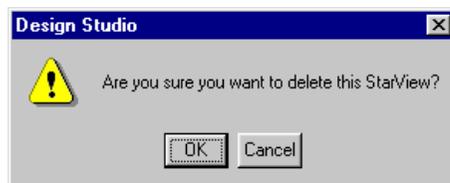
StarView から論理ディメンションが削除されます。

StarView の削除

StarView を削除すると、リポジトリのオブジェクト定義が削除されます。BaseView の表定義、および基礎となるデータベースの物理表は影響されません。

StarView を削除するには、次の手順を実行します。

1. BaseView エディタで、BaseView ドロップダウン・リストから関連する BaseView の名前を選択します。
2. BaseView 表リストで、リスト上端の上下の矢印をドラックして、StarView エディタを表示します。
3. 「StarView」で右クリックして、「Delete」を選択します。
StarView ウィザードでは、削除の確認を求められます。



4. 「OK」をクリックして StarView を削除します。

StarView は削除され、プランの実行時にエンド・ユーザーは、この StarView の集計表にアクセスしません。

注意： StarView は、BaseView に依存しています。BaseView を削除すると、関連する StarView も削除されます。

集計の環境のメンテナンス

StarMart の構築後、StarMart を最新に保持するには、メンテナンス・タスクを実行する必要があります。

ここで説明する内容は次のとおりです。

- [集計表のリフレッシュ](#)
- [移入のスケジューリング](#)
- [集計表の統計のリフレッシュ](#)
- [集計ポートフォリオの調整](#)
- [既存の集計表の使用](#)
- [集計移入プランのカスタマイズ](#)

集計表のリフレッシュ

最初に StarMart を構築すると、デフォルトでは StarView ウィザードによってプランが作成され、新しい集計表が移入されます。このプランでは、集計ファクト表およびディメンション表が存在する場合、それらすべてを削除し、ベース・レベル・ディメンション表の値をディメンション表に移入してから、ファクト・レコードを集計し、それらを集計ファクト表に挿入します。集計表の最初のロードには、これらの操作が必要です。

集計表の増分ロードでは、ベース・レベル表に対する変更を追跡すると、移入を簡素化できます。StarView エディタの右クリック・メニューを使用すると、StarView ウィザードで、集計表のサブセットを移入するプランを作成できます。たとえば、ディメンション情報に変更されていないことが分かっている場合、ベース・レベル・ファクト表の名前を右クリックして、この表から作成されたすべての集計ファクト表に移入するプランを作成できます。ディメンション表は更新されません。同様に、1つのディメンション表が変更され、ファクト全体が新しいことが分かっている場合は、そのベース・レベル・ディメンション表をクリックして、ディメンション表と関連するすべてのファクト表に移入します。

移入プランによって影響を受ける表は、「Populate」の選択時に選択される StarView アイテムによって決まります。StarView ウィザードでは、StarView の次のどの表の組合せでも移入できるプランを作成できます。

選択された StarView アイテム	プランによって移入される表
 StarView	すべてのディメンションのすべての表および依存ファクト表
 論理ファクト	関連ファクト表
 ファクト表	論理ファクトに関連付けられるすべてのファクト表
 ディメンション表	ディメンション表および関連ファクト表
 ディメンション・コンテナ	すべてのディメンションのすべての表および関連ファクト表

集計移入プランを作成するには、次の手順を実行します。

1. 移入する表のセットの集計階層に該当するアイテムをクリックします。
2. 右クリックして、「Populate」を選択します。

デフォルトのプランの名前を含むダイアログ・ボックスが表示されます。この名前を使用するか、または異なる名前を入力できます。



3. 「OK」をクリックします。

StarView ウィザードでは、集計移入プランを Plan Bin に保存します。ユーザーはこのプランの所有者になります。集計表に移入するには、都合のよい時間にプランを実行するようにスケジュールします。ただちにプランを実行することもできますが、集計の作成は多くのリソースおよび時間を必要とするプロセスになる場合があります。ほとんどの場合、他のデータベース・アクティビティが最小のときに、データをロードしてから、プランを実行するようにスケジュールする必要があります。

注意： StarView と基礎となる BaseView またはデータベースの間にアクティブなリンクは存在しません。新しいデータがベース・レベル表にロードされると、集計表のデータをリフレッシュする必要があります。新しいデータをベース・レベル表にロードしても、集計表の更新に失敗する場合、クライアント・アプリケーションに表示される集計データは古いデータになります。

移入のスケジュールリング

Design Studio のスケジューラを使用して、自動的に集計表を移入します。

移入プランをスケジュールするには、次の手順を実行します。

1. Design Studio で、標準ツールバーの「Scheduler」ボタンをクリックします。
「Scheduler」ダイアログ・ボックスには、カレンダーおよびタイム・スロットが表示されます。現在の日時が選択されます。
2. カレンダーをクリックして、移入プランを実行する日付を選択します。
3. 右側の時間のリストをスクロールして、プランをスケジュールする時間を表示します。多くの場合、サーバー・システムを使用するユーザーがいない夜間になります。異なる時間を指定しない場合は、現在の時間が選択されます。
4. Plan Bin の移入プランを、プランを実行する時間にドラッグします。時間をクリックして、選択されていることを確認します。これにより、ドロップダウン・リストで、頻度、通知および保存オプションが有効になります。

5. 「frequency」ドロップダウン・リストで、頻度を選択して集計表を再移入します。集計ファクト表の場合、これは、ベース・レベル表、または集計ファクト表の単位と同じ頻度である必要があります。カレンダーのスケジュールされているすべての曜日に、時計アイコンが表示されます。
6. プラン実行時の通知方法のオプションを選択します。通知方法として電子メール、ログ・ファイル、その両方を選択できます。選択しないこともできます。
7. 「OK」をクリックして変更を保存します。

Plan Bin の集計移入プランのアイコンに時計が表示され、スケジュールされていることを示します。

集計表の統計のリフレッシュ

集計の環境のメンテナンスの重要な部分は、集計表が使用されていることを確認することです。StarView エディタの「Fact and Dimension Table Properties」ダイアログ・ボックスでは、集計表の統計が表示されます。これらの統計には、表が Oracle BAM Enterprise Link クライアント・アプリケーションからアクセスされた回数、およびレコード単位の表サイズが含まれます。この情報を取得するには、Oracle BAM Enterprise Link アカウントにマップされたデータベース・ログインが、表に対して選択を実行できる必要があります。

注意： 集計表に移入するたびに、集計表の統計をリフレッシュする必要があります。統計をリフレッシュすると、StarView ウィザードでこれらの表へのナビゲートに使用するメタデータが更新されます。

データベース表に対して SQL COUNT 関数を実行して、表サイズ（レコード・カウント）を取得します。集計ナビゲーション時に、StarView ウィザードではこの数を使用して、プランに使用できる最小の表を決定します。

表がアクセスされた回数（ヒット・カウント）は、Oracle BAM Enterprise Link クライアント・アプリケーションのみからのアクセスを示します。他のインタフェースからのアクセスは含みません。この情報は、リポジトリに格納されます。ヒット・カウントを使用して、ユーザーが表にアクセスしているかどうかを判断できます。

StarView エディタで集計表の統計をリフレッシュするには、次の手順を実行します。

1. 「StarView」をクリックします。
2. 右クリックして、「Refresh Statistics」を選択します。

集計ポートフォリオの調整

物理的な集計表は、StarView または BaseView の表のメタデータとは異なります。StarView から集計表を削除しても、これらの集計表はまだデータベースに存在します。ただし、集計ナビゲーションの StarView ウィザードは、それらの集計表にアクセスできません。

StarView の表を削除して再作成するか、または完全に削除するには、StarView、BaseView および物理的なデータベースからそれらを削除します。StarView に新しい表を作成しようとして、その表がすでにデータベースに存在する場合は、エラーが発生します。

既存の集計表の使用

既存の集計表を所有しており、Oracle BAM Enterprise Link クライアント・アプリケーションからそれらにナビゲートする場合、StarView を作成してサマリー表を StarView に関連付けることができます。

次に、既存の集計表の StarView を作成するプロセスについて概要を示します。

1. 既存の集計表と関連付けられるベース・レベル表の BaseView を作成します。
2. BaseView の StarView を作成します。手順については、[134 ページの「StarView の作成」](#)を参照してください。
3. 1 つ以上のベース・レベル・ファクト表、集計するメジャー、および適用する集計関数の指定を含む各ファクトを定義します。手順については、[146 ページの「ファクトへのメジャーの追加」](#)を参照してください。
ファクトに新しいメジャーを割り当てる際は、既存の集計ファクト表で、そのメジャーの名前を使用します。
4. 1 つ以上のベース・レベル・ディメンション表、属性階層の指定など、各ディメンションを定義します。手順については、[151 ページの「StarView へのディメンションの追加」](#)を参照してください。
ディメンションに新しい属性を追加する際は、既存の集計ディメンション表でその属性の名前を使用します。
5. 集計表を作成します。手順については、[141 ページの「集計表の作成」](#)を参照してください。
6. 既存の集計表のデータベース・ビューを作成します。ビューに名前を付ける際は、StarView の集計表の名前を使用します。
7. BaseView エディタで、新しい集計表を、集計表のビューと置換します。

StarView ウィザードでは、物理的な表と同様に、既存の集計表のビューにナビゲートします。

集計移入プランのカスタマイズ

集計移入プランをカスタマイズして、プランの実行と同時に他のデータベース操作を実行できます。たとえば、このプランに索引メンテナンスを含むことがあります。ベース・レベル表で変更された情報がわかっている場合、集計表を移入するためにプランの SQL をカスタマイズできます。ベース・レベル表から新しい情報のみを抽出する文を追加し、それを集計表にロードできます。

集計移入プランをカスタマイズするには、次の手順を実行します。

1. **Design Studio** で、**Plan Bin** からワークスペースに移入プランをドラックします。
2. データ・フロー・ハンドルをドラックして、データ・フロー・エディタを表示します。
3. データ・フロー・エディタで、**SQL Query** ステップを右クリックして、「**SQL Editor**」を選択します。
「**SQL Editor**」ダイアログ・ボックスが表示されます。
4. 追加の **SQL** をダイアログ・ボックスに入力します。
たとえば、**DROP INDEX** 文を追加してから、**Design Studio** で生成された **SELECT** 文、その後に **CREATE INDEX** 文を追加する場合があります。
5. 「**OK**」をクリックして変更を保存します。

メタデータの統合

ここで説明する内容は次のとおりです。

- [メタデータの統合について](#)
- [OLE DB プロバイダとしての Enterprise Link の使用](#)
- [ソースとしての OLE DB プロバイダの使用](#)
- [Microsoft Repository での作業](#)

メタデータの統合について

Design Studio によって、異なる環境間でメタデータを共有するための強力な統合が提供されます。この章では、次のものを使用してメタデータを統合する Design Studio の使用方法について説明します。

- OLE DB プロバイダとしての Oracle BAM Enterprise Link
- OLEDB Source Transform
- Microsoft Repository
- DTS 変換

OLE DB プロバイダとしての Enterprise Link の使用

OLE DB プロバイダとして Oracle BAM Enterprise Link を使用し、他のアプリケーションからリポジトリのプランにアクセスできます。OLE DB とは、アプリケーションに、異なるタイプのソースに格納されているデータへのアクセスを提供する Component Object Model (COM) インタフェース群です。OLE DB コンシューマは、プロバイダによって公開されているデータにアクセスします。Enterprise Link OLE DB プロバイダの名前は SaIdc です。OLE DB は COM 仕様であるため、プログラマは、OLE DB データ・ソースを使用するクライアントを作成する C++ コードを書くことができます。

注意： Microsoft プラットフォーム上でのみ OLE DB はサポートされます。

ActiveX Data Objects (ADO) を経由してリポジトリの情報に接続することもできます。ADO は、言語非依存の OLE DB コンシューマです。ADO を使用して、ASP ページ、Java アプレット、および Visual Basic for Applications (VBA) を使用するデータを表示できます。

Enterprise Link 情報との ADO 接続の例には、Microsoft Excel スプレッドシートにプランの結果を表示する VBA コードがあります。Enterprise Link プロバイダへの接続、および Design Studio のユーザー・ログイン情報を含む VBA コードを書くことができます。SQL テキストで、プランおよびシンクを指定できます。コード例に、ADO を使用して Excel からプランを実行するように書く必要がある VBA コードの一部を示します。Design Studio とインストールされている ADO 2.0 を所有する必要があります。

次に示すのは、Enterprise Link に接続してプランを選択する VBA コードです。

```
Private Sub Connect()
'Establish connection with the server
  If con Is Nothing Then
    Set con = CreateObject("ADODB.Connection")
    con.Open "provider=SaIdc;UserID=gthompson;Password=;"
    MsgBox "Logged In"
  End If
End Sub
Private Sub FillInArray()
'Fill in all the commands we wanted to run in an array
  SqlText(0) = "select * from ""Cost of Sales Plan,
created 2/22/99""
where sink = 'Grid'"
End Sub
```

OLE DB コンシューマを Enterprise Link メタデータに接続するには、MERANT（以前は Intersolv として知られていた）製の Connect ODBC/OLE DB Adapter など、ODBC/OLE DB ブリッジを提供するサードパーティの ODBC ドライバを使用することもできます。Enterprise Link メタデータは、リポジトリ・データベースの表として示されるプランおよびシンクで構成されます。コンシューマ・アプリケーションに応じて、プランのリストおよびプラン:シンク形式のコンテンツを表示できます。プランの各シンクは、表として表されます。シンクの各フィールドは、列として表されます。たとえば、Q4Sales というプランに 2 つの Grid が含まれる場合、各シンクは Q4Sales:Grid および Q4Sales:Grid2 という表で表されます。各表には、Manager、Store、Sales_in_Dollars などのシンクで使用されるフィールドが含まれます。

最初に、Enterprise OLE DB プロバイダ、SaIdc への ODBC データ・ソースとして Connect ODBC/OLE DB Adapter をインストールおよび構成する必要があります。その後、構成した Connect ODBC/OLE DB Adapter を指定して、OLE DB コンシューマ・アプリケーションからリポジトリに接続します。

ソースとしての OLE DB プロバイダの使用

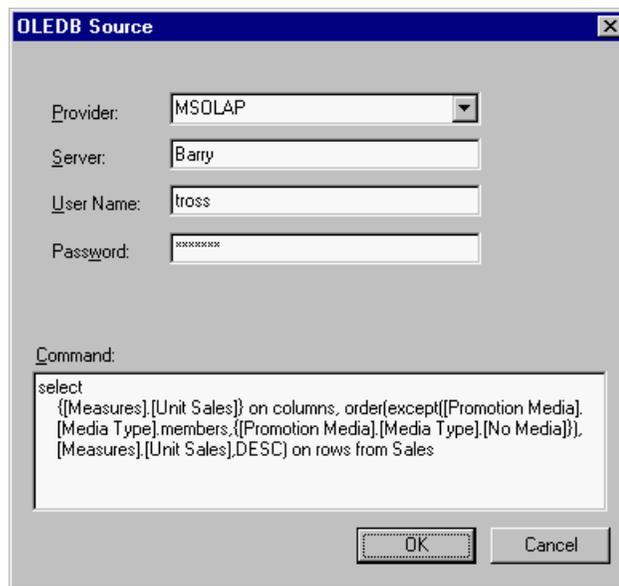
OLEDB Source Transform では、OLE DB プロバイダがデータ・フロー内のデータ・ソースとして使用されます。OLEDB Source Transform は、OLE DB または OLAP インタフェース用の OLE DB が提供されているソースにアクセスします。この Transform では、他のソースからデータを取得するための MDX および SQL コマンドの両方がサポートされています。MDX は、OLAP 用の OLE DB の多次元問合せ構文で、OLAP データ・ソースを問い合わせるための SQL 言語の上に定義されているマクロで構成されています。OLEDB Source Transform では、いくつかのタイプの単純なプロバイダまたは複雑なプロバイダを使用できます。

注意： Microsoft プラットフォーム上でのみ OLE DB はサポートされます。

プロバイダ名をダイアログ・ボックスのドロップダウン・リストに表示するには、OLE DB プロバイダをクライアント・マシンおよび Enterprise Link Server の両方にインストールする必要があります。プロバイダが Enterprise Link Server にのみインストールされている場合は、ダイアログ・ボックスでそのプロバイダの名前を入力する必要があります。メタデータが含まれているサーバーをプロバイダとして使用するには、そのサーバーを実行している必要があります。他のプロバイダの例としては、SQL Server 用の SQLOLEDB、ODBC 接続用の MSDASQL などがあります。

OLEDB Source Transform を使用するには、次の手順を実行します。

1. OLEDB Source をデータ・フローに配置し、そのダイアログ・ボックスを表示します。
2. プロバイダをドロップダウン・リストから選択します。プロバイダが Enterprise Link Server にのみインストールされている場合は、このフィールドにそのプロバイダの名前を入力します。
3. アクセスするサーバーの名前を入力します。
4. ユーザー名およびパスワードを入力します。
5. 実行する MDX または SQL コマンドを入力します。次のダイアログ・ボックスは、単純な MDX の例です。



6. 「OK」をクリックして、ダイアログ・ボックスを閉じます。

ユーザー入力の SQL と同様に文が処理されるため、データ・フローで OLEDB Source Transform の後に続く Transform 内の列名は表示されません。データ・フローで追加の処理を行うために他の Transform 内の列名を入力できます。

Microsoft Repository での作業

Design Studio によって、BaseView エディタを使用して BaseView をエクスポートおよびインポートするために Microsoft Repository との統合が提供されます。Design Studio では、ワークスペースのプランとして Microsoft Repository に格納されている Microsoft DTS 変換をインポートすることもできます。「Export BaseView」ダイアログ・ボックスなどの Design Studio のダイアログ・ボックスでは、Microsoft Repository のデータ構造の階層が表示されます。BaseView をエクスポートする際に、表および列の Enterprise Link メタデータ・ビューを Microsoft Repository の階層にエクスポートしています。

ここで説明する内容は次のとおりです。

- BaseViews のエクスポート
- Microsoft Repository オブジェクト階層
- Microsoft Repository からの変換のインポート
- DTS-Scripting Transform の使用

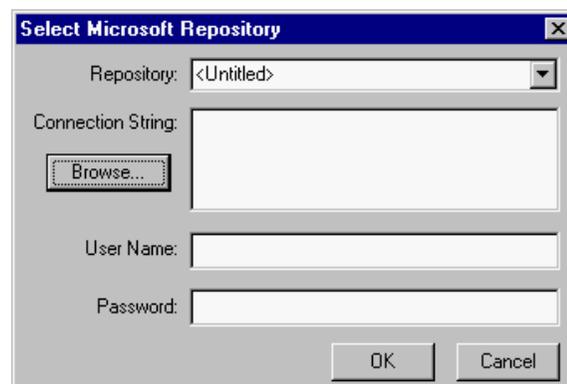
BaseViews のエクスポート

BaseView を Microsoft Repository のスキーマ・オブジェクトにエクスポートできます。Microsoft Repository からの BaseView のインポートの詳細は、[32 ページの「Microsoft Repository を使用した BaseView の作成」](#)を参照してください。

BaseView をエクスポートするには、次の手順を実行します。

1. BaseView エディタで、BaseView を選択して「BaseView」ドロップダウン・リストからエクスポートします。
2. 「Export BaseView」ボタンをクリックします。

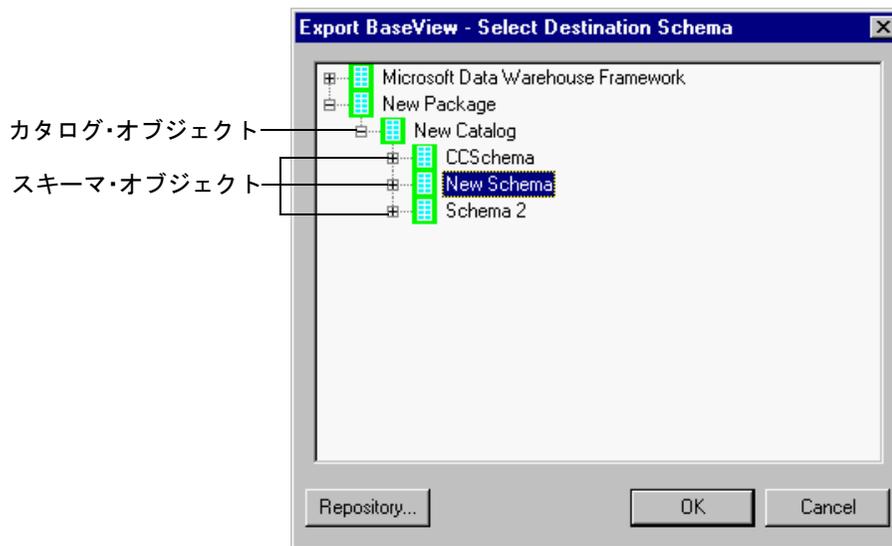
「Select Microsoft Repository」ダイアログ・ボックスが表示されます。



- Microsoft Repository のログイン情報を入力して、「OK」をクリックします。

- **Repository:** Microsoft Repository の名前。
- **Connection String:** Microsoft Repository の接続文字列。データベースが SQL Server の場合は、`server=servername;database=msdb` という書式を使用します。データベースが Microsoft Access ファイルの場合は、「Browse」ボタンをクリックし、そのファイルの場所を特定して、`C:\¥access.mdb` という書式を使用します。
- **User Name:** データベース・ユーザーの名前。
- **Password:** ユーザー・アカウントのパスワード。

「Select Destination Schema」ダイアログ・ボックスが表示されます。Microsoft Repository にまだオブジェクトを作成していない場合、このダイアログ・ボックスは空です。



- ダイアログ・ボックスを右クリックして、メニューからオブジェクトの作成を選択します。オブジェクト階層を様々な方法で作成できます。たとえば、パッケージを作成した後にカタログを作成し、その後でスキーマを作成します。
- BaseView をエクスポートするスキーマ・オブジェクトを選択します。
- 「OK」をクリックします。

BaseView がエクスポートされます。

Microsoft Repository から、メタデータを Visio 5.0 Enterprise Edition、ERwin、Visual Modeler などのモデリング・ツールにエクスポートできます。変更後、BaseView を Design Studio に再度インポートするために、新しい BaseView を作成するか、または BaseView を同期化できます。

Microsoft Repository オブジェクト階層

Microsoft Repository で、メタデータ・オブジェクトは階層に編成されます。カタログおよびスキーマ・オブジェクトを含むオブジェクトの階層を作成する必要があります。階層はルート・レベルでカタログ・オブジェクトで開始するか、またはシステムで開始して、階層の各レベルを含むことができます。BaseView をエクスポートしているときに Design Studio でこの階層を作成および編集できます。次の表にオブジェクト階層が示されます。

オブジェクト	説明
システム	ルート・レベル
モデル	第 2 レベル
データソース	第 3 レベル（通常は使用されません）
パッケージ	第 3 レベル。オブジェクト・タイプに応じて、タスクまたはカタログを含みます。
タスク	第 4 レベル。プランの定義のレベルです。
カタログ	第 4 レベル
スキーマ	BaseView のレベル。表および列を含みます。

詳細は、Microsoft Repository ドキュメントを参照してください。

Microsoft Repository からの変換のインポート

Microsoft Repository から Microsoft DTS 変換をインポートできます。変換は、Microsoft SQL Server 7.0 以上の DTS ウィザードを使用して作成されるタスクのタイプです。

変換をインポートすると、Design Studio は DTS-Scripting Transform を含むデータ・フローを自動的に作成します。

Microsoft Repository および SQL Server 7.0 がインストールされていない場合でも、プランで DTS-Scripting Transform を使用して、JScript または VBScript をデータ・フローに挿入できます。詳細は、[173 ページの「独自のスクリプトの作成」](#)を参照してください。

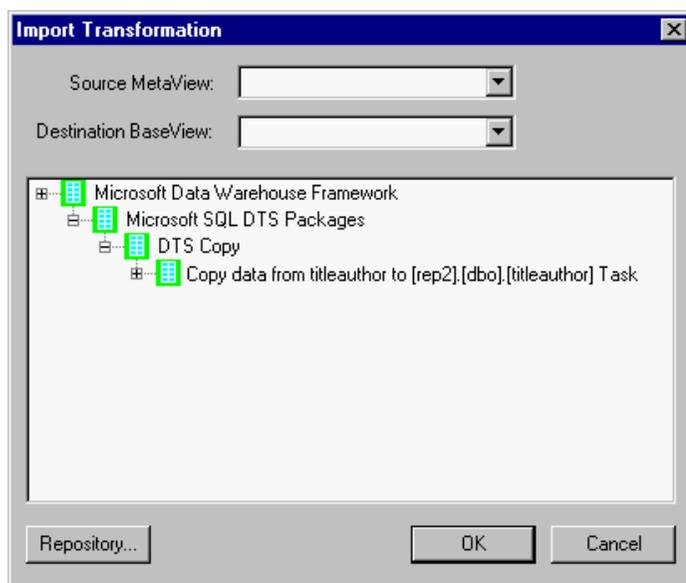
Microsoft Repository から変換をインポートするには、次の手順を実行します。

1. Design Studio のメニュー・バーで、「File」→「Import Plan Definition」→「From Microsoft Repository」を選択します。

「Select Microsoft Repository」ダイアログ・ボックスが表示されます。

2. Microsoft Repository のログイン情報を入力して、「OK」をクリックします。
 - **Repository:** Microsoft Repository の名前。
 - **Connection String:** Microsoft Repository の接続文字列。データベースが SQL Server の場合は、server=servername;database=msdb という書式を使用します。データベースが Microsoft Access ファイルの場合は、「Browse」ボタンをクリックし、そのファイルの場所を特定して、C:¥access.mdb という書式を使用します。
 - **User Name:** データベース・ユーザーの名前。
 - **Password:** ユーザー・アカウントのパスワード。

「Import Transformation」ダイアログ・ボックスが表示されます。



3. ドロップダウン・リストから「Source MetaView」を選択します。ソース MetaView は、プランで使用される Part のソースです。
4. ドロップダウン・リストから「Destination BaseView」を選択します。宛先 BaseView は、新しいメタデータのターゲットです。
5. ツリー・コントロールから変換オブジェクトを選択します。
6. 「OK」をクリックします。

SQL Query、DTS-Scripting ステップおよび Microsoft Batch Loader を含むデータ・フロー・プランが作成されます。SQL Query では、指定した MetaView から Part が選択されます。Microsoft Batch Loader が、指定した BaseView に列をロードするように、Microsoft Batch Loader の設定を指定する必要があります。DTS 変換のソース列名と宛先列名が一致しない場合、Rename Column ステップがプランに追加されます。

7. データ・フロー内の各ステップの設定を指定します。DTS-Scripting Transform の詳細は、171 ページの「[DTS-Scripting Transform の使用](#)」を参照してください。

DTS-Scripting Transform の使用

プランの定義をインポートしたり、独自のスクリプトをプランに追加したりするために、DTS-Scripting Transform を使用できます。

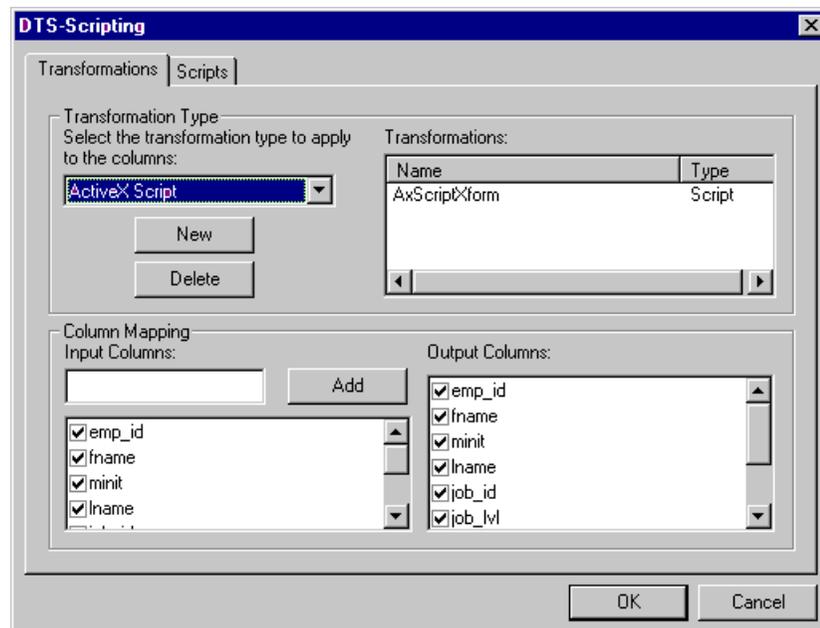
ここで説明する内容は次のとおりです。

- プランの定義のインポート
- 独自のスクリプトの作成

プランの定義のインポート

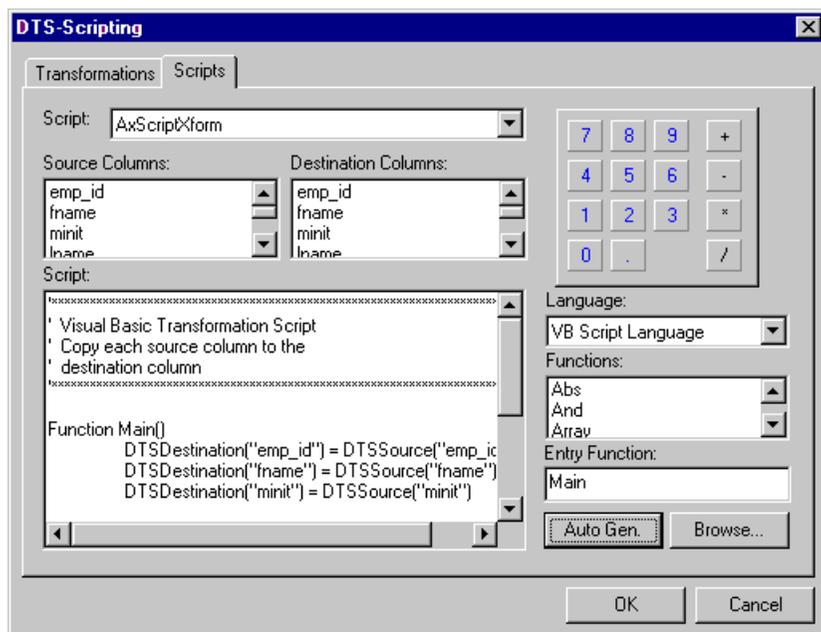
DTS-Scripting Transform を使用するには、次の手順を使用します。

1. プランの定義をインポートしてデータ・フローを表示してから、「DTS-Scripting Transform」ダイアログ・ボックスを開きます。



2. 「Input Columns」と「Output Columns」リストでマップする列のチェック・ボックスを選択します。

3. 「**Scripts**」 タブをクリックします。変換タイプが **Copy** の場合、スクリプトを追加または編集することはできません。



4. 次のどの方法を使用してもスクリプトを編集できます。
 - スクリプト言語として **VBScript** または **JScript** を選択し、「**Auto Gen**」をクリックしてスクリプトを生成します。
 - スクリプトを直接入力します。
 - ソースまたは宛先リストの列名をダブルクリックして、それをスクリプトに追加します。
 - カルキュレータの演算子および数値ボタンをクリックします。
 - **VBScript** の関数を使用します。
 - 「**Auto Gen**」をクリックしてスクリプトをリセットします。
5. 「**OK**」をクリックしてダイアログ・ボックスを閉じます。
6. 「**Update**」をクリックして、プランを実行します。

独自のスクリプトの作成

DTS-Scripting Transform をプランに追加して、VBScript または JScript のスクリプトを追加できます。スクリプト言語を参照するには、Web 上の Microsoft または別のベンダーのドキュメントを参照します。

DTS-Scripting Transform を使用してプランにスクリプトを追加するには、次の手順を使用します。

1. DTS-Scripting Transform をデータ・フローに配置し、そのダイアログ・ボックスを表示します。
「DTS-Scripting Transform」ダイアログ・ボックスが表示されます。
2. 変換タイプとして「**ActiveX Script**」を選択し、「**New**」をクリックします。
変換は、右側の変換のリストに追加されます。
3. リスト内の変換を選択して、スクリプトに含める入力列および出力列をチェックします。
4. 「**Scripts**」タブをクリックします。
5. 「**Script**」ドロップダウンから変換を選択します。
6. 使用する言語を選択します。
7. 「**Auto Gen**」をクリックして、選択した言語でスクリプトを生成します。
8. 次のすべてを使用してスクリプトを編集できます。
 - スクリプト言語として VBScript または JScript を選択し、「**Auto Gen**」をクリックしてスクリプトを生成します。
 - スクリプトを直接入力します。
 - ソースまたは宛先リストの列名をダブルクリックして、それをスクリプトに追加します。
 - カルキュレータの演算子および数値ボタンをクリックします。
 - VBScript の関数を使用します。
 - 「**Auto Gen**」をクリックしてスクリプトをリセットします。
9. 「**OK**」をクリックします。

VBScript Transform

ここで説明する内容は次のとおりです。

- [Visual Basic Script Transform](#)
- [ActiveX Automation のメソッドおよびイベント](#)
- [VBScript のエラー処理](#)
- [カスタム VBScript の例](#)

Visual Basic Script Transform

Microsoft Visual Basic Scripting Edition (VBScript) は、サーバー・システム用に最適化されている Microsoft Visual Basic プログラミング・システムのサブセットです。VBScript エンジンは、Oracle BAM Enterprise Link に組み込まれています。

Oracle BAM Enterprise Link 環境では、VBScript を使用してレコード・レベルでデータを操作できます。SQL とは異なり、VBScript は一度に 1 レコードを処理します。このため、ユーザーは結果セットの各レコードの状態を変更できます。VBScript は Enterprise Link Server 上で実行されるため、対話型操作を実行するためでなく、サーバー上でレコードを処理するために VBScript Transform を使用します。たとえば、VBScript Transform では MsgBox または InputBox を実装できません。

VBScript の Enterprise Link 実装では、別のアプリケーションから複合ドキュメント内にアイテムを格納するリンクがサポートされます。VBScript Transform では、ActiveX 自動化サーバーとのデータの読書きが可能です。VBScript Transform では、別のアプリケーションに属している公開オブジェクトを操作することはできません。

ここで説明する内容は次のとおりです。

- [VBScript Transform](#)
- [VBScript Transform の使用](#)

VBScript Transform

Design Studio には、データ・フロー・プランのカスタム・ソース、ステップおよびシンクを作成するための次の VBScript Transform が含まれます。

■ VBScriptCopy

この Transform は、入力レコードをコピーし、入力レコードのデータおよびその列属性を操作できます。列を追加または削除するか、あるいはサイズ、タイプ、名前などの列属性を変更する場合、この Transform を使用します。

■ VBScriptInplace

この Transform は、各レコードをインプレース処理します。そのため、入力レコードのデータのみを変更できます。列の追加または削除、あるいは列属性の変更は実行できません。たとえば、10 文字の文字列が含まれている入力レコード内の列は、10 文字を超えて出力することはできません。

■ VBScriptSource

この Transform は、外部アプリケーションから入力データを読み取ります。ソース・アプリケーションは、OLE 準拠で、クライアントと Enterprise Link Server の両方にインストールされている必要があります。

■ VBScriptSink

この Transform では、レコードが外部アプリケーション（Microsoft Excel、Microsoft Word など）に出力されます。ターゲット・アプリケーションは、OLE 準拠で、クライアント・マシンと Enterprise Link Server の両方にインストールされている必要があります。

VBScript Transform の使用

プランで使用する VBScript Transform のタイプを決定した後、カスタム VBScript Transform の作成は、3 つのステップで構成されます。

1. VBScript エディタで Visual Basic スクリプトを入力します。
2. スクリプトを送信して実行します。
3. プランを使用するか、または Tool Bin のカスタム Transform として、VBScript ステップを保存します。

ここで説明する内容は次のとおりです。

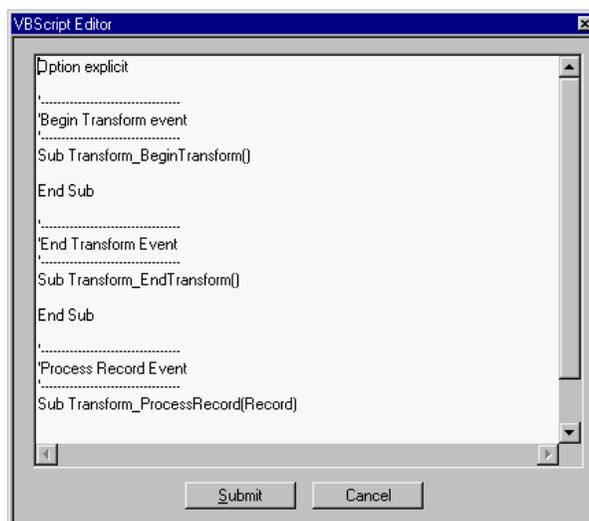
- [VBScript エディタ](#)
- [カスタム VBScript Transform の保存](#)

VBScript エディタ

VBScript エディタで Visual Basic コードを入力して、カスタム VBScript Transform を作成します。VBScript エディタでは、空のイベント・プロシージャで構成されるテンプレートが表示されます。これらのイベント・プロシージャについては、[180 ページの「OLE イベント・プロシージャ」](#)を参照してください。

VBScript エディタを開くには、次のようにします。

- データ・フロー内の VBScript ステップをダブルクリックします。



VBScript エディタを使用して、プランで実行するためにスクリプトを入力して送信します。

- スクリプトをコンパイルして Data Flow Service に送信するには、「**Submit**」をクリックします。

「Submit」をクリックすると、VBScript エンジンがスクリプトを解析します。エラーが検出された場合、エラー・ダイアログ・ボックスが表示されます。エラーが発生しない場合、プランの実行時にスクリプトは Data Flow Service に送信され実行されます。

「Submit」をクリックすると、スクリプトは保存され VBScript エディタは閉じます。

- VBScript エディタを閉じて変更を破棄するには、「**Cancel**」をクリックします。

カスタム VBScript Transform の保存

プランの一部、または Tool Bin の Transform として、カスタム VBScript Transform を保存できます。

Tool Bin にカスタム VBScript Transform を保存するには、次の手順を実行します。

1. データ・フローの VBScript ステップを右クリックし、「**Create Transform**」を選択します。
2. Transform の名前を編集し、説明およびヒント・テキストを追加します。Information Studio ユーザーがこの Transform を参照しないようにするには、アクセシビリティを「**Design Studio only**」に設定します。
3. 「**OK**」をクリックして、Transform を Tool Bin に追加します。Transform は、元の Transform を選択したカテゴリと同じカテゴリに追加されます。

カスタム Transform を削除するには、Tool Bin で選択して [Delete] キーを押します。Tool Bin にアクセスできる他のユーザーは、カスタム Transform にアクセスできません。カスタム Transform は、所有者によってのみ削除できます。

ActiveX Automation のメソッドおよびイベント

VBScript は、ActiveX Automation を介してホスト・アプリケーションと動作します。Oracle BAM Enterprise Link の VBScript コンポーネントは、イベントを実装しメソッドを公開します。ActiveX の場合、イベントは外部アクションによってトリガーされる送信インタフェースで、メソッドはプログラムによってコールされる受信インタフェースです。

この項では、データ操作をカスタマイズするために VBScript Transform で使用できるイベントおよびメソッドのカテゴリについて説明します。

- [VBScript Transform イベント](#)
- [OLE イベント・プロシージャ](#)
- [レコード・オブジェクト API](#)
- [RecordDescriptor Object API](#)
- [ColumnDescriptor Object API](#)
- [ActiveX クライアント・サポート](#)
- [その他の API](#)
- [自動化サーバーの作成](#)
- [VBScript のフィールド・データ型](#)

VBScript Transform イベント

次の表に、各 VBScript Transform によって使用されるイベント・プロシージャを示します。

VBScript Transform	使用されるイベント
VBScriptInplace	Transform_BeginTransform() Transform_EndTransform() Transform_ProcessRecord(Record)
VBScriptCopy	Transform_BeginTransform() Transform_EndTransform() Transform_ProcessRecordCopy(RecIn, RecOut) Transform_CalculateSchema(RecDesc)
VBScriptSource	Transform_BeginTransform() Transform_EndTransform() Transform_ProcessRecord(Record) Transform_CalculateSchema(RecDesc)
VBScriptSink	Transform_BeginTransform() Transform_EndTransform() Transform_ProcessRecord(Record)

OLE イベント・プロシージャ

VBScript Transform では、次のイベント・プロシージャをサポートします。

Transform_BeginTransform()

説明：

最初にコールされるイベント。このプロシージャに初期化コードを配置します。

例：

```
Sub Transform_BeginTransform()  
End Sub
```

Transform_EndTransform()

説明：

最後にコールされるイベント。このプロシージャにコードを配置し、作業または他のタスクをクリーンアップします。

例：

```
Sub Transform_EndTransform()  
End Sub
```

Transform_CalculateSchema()

説明：

このイベントは、BeginTransform の後にコールされます。このプロシージャにコードを配置して、RecordDescriptor および ColumnDescriptor を変更します。

パラメータ：

変更する RecordDescriptor を参照します。

例：

```
Sub Transform_CalculateSchema(RecDesc)  
End Sub
```

Transform_ProcessRecord()

説明：

VBScriptInplace Transform のイベント・プロシージャを操作する主要なレコード。このプロシージャには、レコード関連のタスクを実行するコードを配置します。

パラメータ：

現在、選択されているレコードを参照します。

例：

```
Sub Transform_ProcessRecord(Record)  
End Sub
```

Transform_ProcessRecordCopy()

説明：

VBScriptCopy Transform のイベント・プロシージャを操作する主要なレコード。レコード関連のタスクを実行するコードを配置します。

パラメータ：

- 元の入力レコードを参照します。それは、読取り専用としてマークされます。すべての Set* 演算は、例外をトリガーします。
- 出力レコード、読取り / 書込みレコード。これは、変更されたレコードへの参照です。

例：

```
Sub Transform_ProcessRecordCopy(RecordIn,RecordOut)
End Sub
```

レコード・オブジェクト API

VBScript エンジンには、一度に 1 レコード・オブジェクトを処理します。レコード・オブジェクトは、データベースから取得されるデータの行を表します。レコード・オブジェクトでは、次のメソッドをサポートします。

GetFieldValue

説明：

フィールドの値を取得するコール。

パラメータ：

- 取得するレコードのフィールド数。
- 指定したフィールドの返される値。

例：

```
debt = recordin.getfieldvalue(2)
```

SetFieldIndex

説明：

レコードにアクティブ・フィールドを設定するコール。

パラメータ：

フィールドに設定する索引の値。

例：

```
recordout.setfieldindex = 3
```

CountFields

説明：

レコードのフィールド数を取得するコール。

パラメータ：

現在のレコードに返されるフィールド数。

例：

```
fldcount = recordin.countfields
```

SetFieldValue

説明：

フィールドの新しい値を設定するコール。このコールの前に SetFieldIndex() が存在する必要があります。

パラメータ：

現行フィールドに設定する新しい値。

例：

```
recordout.setfieldindex = 1  
recordout.setfieldvalue = ratio
```

RecordDescriptor Object API

RecordDescriptor オブジェクトでは、次のメソッドをサポートします。

GetColumn

説明：

レコードの指定された列にアクセスするコール。

パラメータ：

- 取得するレコードの列数 (0 ベース)。
- 返される列オブジェクトのインスタンスへの参照。

例：

```
set col = recdesc.getcolumn(0)
```

CreateColumn

説明：

レコードに新しい列を作成するコール。新しい列は、レコード内の列のリストの最後に追加されます。ColumnDescriptor メソッドを使用して、列を設定および変更します。

パラメータ：

返される列オブジェクトのインスタンスへの参照。

例：

```
set col = recdesc.createcolumn
```

DeleteColumn

説明：

指定した列をレコードから削除するコール (0 ベース)。

パラメータ：

削除する列の位置。

例：

```
delcol = recdesc.deletecolumn(2)
```

CountColumns

説明：

レコードの列数を取得するコール。

パラメータ：

現行レコード内の列数。

例：

```
colcount = recdesc.countcolumns
```

ColumnDescriptor Object API

次のメソッドは、RecordDescriptor API GetColumn および CreateColumn メソッドから返されるオブジェクトと動作します。

SetColumnName

説明：

新しい列に名前を設定するコール。

パラメータ：

現行の列の新しい名前。

例：

```
col.setcolumnname = "Debt/Income"
```

SetColumnDataType

説明：

新しい列にデータ型を設定するコール。

パラメータ：

現行の列の新しいデータ型。サポートされているデータ型を示す値リストについては、[187 ページの「VBScript のフィールド・データ型」](#)を参照してください。新しい列に文字列型を設定する場合、SetColumnMaxSize メソッドを使用して、列のサイズを設定する必要があります。

例：

```
col.setcolumndatatype = 5 'double
```

SetColumnMaxSize

説明：

新しい文字列の列の最大サイズを設定するコール。このメソッドは、固定長データ型に影響しません。このメソッドは、新しい文字列の列のサイズを設定するために使用する必要があります。

パラメータ：

現行の列の最大サイズの値。

例：

```
col.setcolumnmaxsize = 25
```

GetColumnName

説明：

現行の列名を取得するコール。

パラメータ：

現行の列の名前。

例：

```
dim name  
name = col.getcolumnname
```

GetColumnDataType

説明：

列のデータ型を取得するコール。

パラメータ：

現行の列のデータ型の値。サポートされているデータ型を示す値リストについては、[187 ページの「VBScript のフィールド・データ型」](#)を参照してください。

例：

```
dtype = col.getcolumndatatype
```

GetColumnMaxSize

説明：

列の最大サイズを取得するコール。

パラメータ：

現行の列に設定される最大サイズの値。

例：

```
colmax = col.getcolumnmaxsize
```

GetScale

説明：

DECIMAL 列のスケールを取得するコール。

パラメータ：

現行の列のスケールの値。

例：

```
colscale = col.getscale
```

GetPrecision

説明：

DECIMAL 列の精度を取得するコール。

パラメータ：

現行の列の精度の値。

例：

```
colprec = col.getprecision
```

SetScale

説明：

DECIMAL 列のスケールを設定するコール。

パラメータ：

現行の列のスケールの新しい値。

例：

```
col.setscale = 8
```

SetPrecision

説明：

DECIMAL 列の精度を設定するコール。

パラメータ：

現行の列の精度の新しい値。

例：

```
col.setprecision = 4
```

ActiveX クライアント・サポート

CreateObject

説明：

Excel または Word のような自動化サーバーのインスタンスを作成するコール。このサーバーは、Enterprise Link Server 上でインストールして登録する必要があります。= 演算子の左側に、オブジェクト変数と Set VBScript キーワードを使用します。

パラメータ：

- 作成する自動化サーバー・オブジェクトの名前。
- 返される自動化サーバー・オブジェクトのインスタンスへの参照。

例：

```
set xlobj = createobject("Excel.Application")
```

その他の API

EndData

説明：

データの最後に達したことを指定するために、VBScriptSource のこのメソッドをコールします。
この関数のコールに失敗すると、無限ループが発生します。

パラメータ：

入力データの末尾のタイミングを指定します（1 はデータの最後を示します）。

例：

```
if count > 10 then
EndData(1)
end if
```

自動化サーバーの作成

次の手順では、VBScriptSource および VBScriptSink Transform のイベントを使用して自動化サーバーを作成する方法について説明します。

1. BeginTransform イベントに自動化サーバーのインスタンスを作成して初期化します。
2. VBScriptSource Transform の場合、RecordDescriptor Object API および ColumnDescriptor Object API を使用して、CalculateSchema イベント内のステップのスキーマを作成します。
3. 自動化サーバーによって提供される Record Object API とメソッドを使用して、ProcessRecord イベントにイベントを作成します。
4. すべての必要なクリーンアップ作業を実行して、EndTransform イベントの自動化サーバーを非アクティブ化します。

191 ページの「VBScriptSource」および 192 ページの「VBScriptSink」を参照してください。これらの例は、自動化サーバーの作成および使用方法を示しています。

VBScript のフィールド・データ型

次の表に、VBScriptCopy Transform を使用した出力列の作成時に使用できるデータ型を示します。

Enterprise Link の内部定義	データ型	VBScript の値
SADC_DATATP_INT	Integer	1
SADC_DATATP_LONG	Long	2
SADC_DATATP_STRING	String	3
SADC_DATATP_DATETIME	Date/Time	4
SADC_DATATP_DOUBLE	Double	5
SADC_DATATP_UNIT	Unsigned Integer	6
SADC_DATATP_DECIMAL	Decimal	8

VBScript のエラー処理

VBScript のエラーには、コンパイルとランタイムの 2 つのタイプがあります。

ここで説明する内容は次のとおりです。

- [VBScript のコンパイル・エラー](#)
- [VBScript のランタイム・エラー](#)

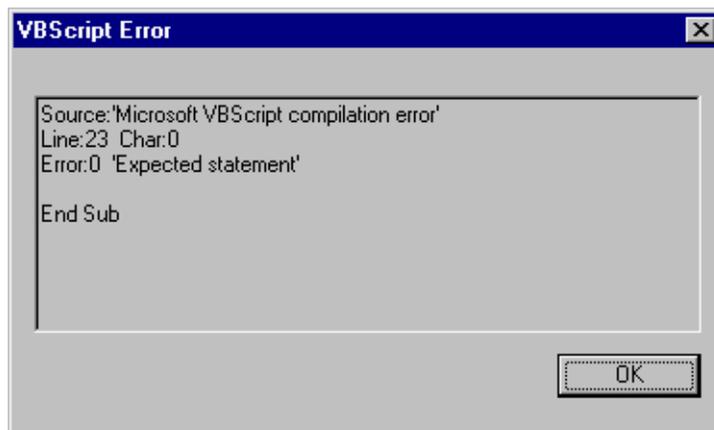
VBScript のコンパイル・エラー

VBScript コードを送信すると、コンパイル・エラーがレポートされます。これらのエラーは、「VBScript Error」ダイアログ・ボックスに表示されます。VBScript エンジンが膨大な行のコードを検索する場合、「VBScript Error」ダイアログ・ボックスを閉じると、VBScript エディタでこのコードが選択されます。

たとえば、次のように Sub 文の前に文字を誤って入力する場合、

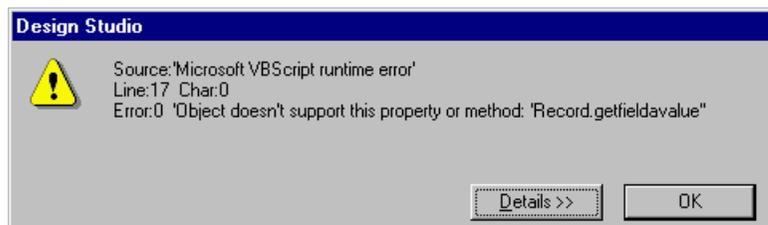
```
aSub Transform_ProcessRecord(Record)
```

VBScript コードをコンパイルすると、次のエラーが発生します。



VBScript のランタイム・エラー

プランを実行する際に、Design Studio エラー・ダイアログ・ボックスでランタイム・エラーがレポートされます。VBScript にスペルの間違ったメソッド名がある場合、プランを実行する際に次のエラーが表示されます。



ダイアログ・ボックスに、VBScript のエラーについての説明とエラーの場所が示されます。ダイアログ・ボックスの「Details」セクションでは、ランタイム・エラーが原因の内部エラーについて説明します。

スペルの間違ったメソッド名によるランタイム・エラーを修正するには、次の手順を実行します。

1. 「OK」をクリックして、エラー・ダイアログ・ボックスを閉じます。
2. VBScript Transform をダブルクリックして、示されたエラーを修正します。
3. 修正したコードを送信します。
4. プランを再実行します。

注意：一度に1つのエラーのみがレポートされます。

カスタム VBScript の例

この項では、カスタム VBScript Transform の例を示します。各 VBScript Transform に対して例が提供されます。

- [VBScriptInplace](#)
- [VBScriptCopy](#)
- [VBScriptSource](#)
- [VBScriptSink](#)

これらの例では、CD で提供される Oracle BAM Enterprise Link のサンプル・データを使用します。

VBScriptInplace

この例では、四半期および年の値を使用して、会計期間の値を作成します。たとえば、四半期の値 2 と 1 年の値 1995 から、会計期間 Q295 が作成されます。この例では、最初の位置に四半期の列、2 番目の位置に年の列、3 番目の位置に任意の文字列の列が含まれるプランが必要です。文字列の列の値は、会計期間の値と置換されます。

```
Option explicit
' requires qtr,year, char(any)
'-----
'Begin Transform event
'-----
Sub Transform_BeginTransform()
End Sub
'-----
'End Transform Event
'-----
Sub Transform_EndTransform()
End Sub
'-----
'Process Record Event
'-----
Sub Transform_ProcessRecord(Record)
dim yyear,qtr
qtr = record.getfieldvalue(0)
yyear = record.getfieldvalue(1)
if yyear = 1994 then yyear = 94
if yyear = 1995 then yyear = 95
record.setfieldindex =2
record.setfieldvalue = yyear & "q" & qtr
End Sub
```

VBScriptCopy

次の例では、負債と収入の比率を計算し、結果を出力する列を作成します。この例では、3つのフィールド（口座または人を識別するフィールド、現在の負債の数値フィールド、および現在の収入の数値フィールド）が含まれるプランが必要です。

```

Option explicit
'outputs (any) and computed debt/income ratio (in created
column)
dim debt, income, ratio, col
'-----
'Begin Transform event
'-----
Sub Transform_BeginTransform()
End Sub
'-----
'End Transform Event
'-----
Sub Transform_EndTransform()
End Sub
'-----
'Process Record Event
'-----
Sub Transform_ProcessRecordCopy (RecordIn, RecordOut)
debt = recordin.getfieldvalue(2)
income = recordin.getfieldvalue(1)
if income = 0 then
ratio = 0
else
ratio = debt/income
end if
recordout.setfieldindex = 3
recordout.setfieldvalue = ratio
recordout.setfieldindex = 0
recordout.setfieldvalue = recordin.getfieldvalue(0)
End Sub
'-----
'Calculate Schema Event
'-----
Sub Transform_CalculateSchema (RecDesc)
set col = recdesc.createcolumn
col.setcolumnname = "Debt/Income"
col.setcolumndatatype = 5 'double
End Sub

```

VBScriptSource

次の例では、Excel ワークシートから 2 列のデータを読み取り、このデータをシンクに出力します。この例を実行するには、ご使用のマシンと Enterprise Link Server に Excel をインストールする必要があります。

```

Option explicit
dim xlobj, col, count
dim a,b,c,d
'-----
'Begin Transform event
'-----
Sub Transform_BeginTransform()
set xlobj = createobject("Excel.Application")
xlobj.workbooks.open("c:\vbs\minifloat2a.xls")
xlobj.visible = true
count = 1
End Sub
'-----
'End Transform Event
'-----
Sub Transform_EndTransform()
xlobj.workbooks.close
xlobj.quit
End Sub
'-----
'Process Record Event
'-----
Sub Transform_ProcessRecord(Record)
a = "A" & cstr(count)
b = "B" & cstr(count)
record.setfieldindex = 0
record.setfieldvalue =
xlobj.workbooks(1).worksheets(1).range(a).value
record.setfieldindex = 1
record.setfieldvalue =
xlobj.workbooks(1).worksheets(1).range(b).value
count = count+1
if count > 10 then
EndData(1)
end if
End Sub
'-----
'Calculate Schema Event
'-----
Sub Transform_CalculateSchema(RecDesc)
set col = recdesc.createcolumn
col.setcolumndatatype = 5 'double
col.setcolumnname = "First"
set col = recdesc.createcolumn
set col = recdesc.getcolumn(1)
col.setcolumndatatype = 5 'double
col.setcolumnname = "Two"
End Sub

```

VBScriptSink

次の例では、3列を Excel ワークシートに出力します。この例では、3列のデータが入力されたデータ・フローが必要です。この例を実行するには、ご使用のクライアント・マシンと Enterprise Link Server に Excel をインストールする必要があります。

```

Option explicit
dim xlobj, col, count
dim a,b,c,d
'-----
'Begin Transform event
'-----
Sub Transform_BeginTransform()
set xlobj = createobject("Excel.Application")
xlobj.workbooks.add
xlobj.visible = true
count = 1
End Sub
'-----
'End Transform Event
'-----
Sub Transform_EndTransform()
xlobj.activesheet.saveas("c:\vbs\vbsnk_3.xls")
'xlobj.workbooks.close()
xlobj.quit
End Sub
'-----
'Process Record Event
'-----
Sub Transform_ProcessRecord(Record)
a = "A"+cstr(count)
b = "B"+cstr(count)
c = "C"+cstr(count)
xlobj.workbooks(1).worksheets(1).range(a).value =
record.getfieldvalue(0)
xlobj.workbooks(1).worksheets(1).range(b).value =
record.getfieldvalue(1)
xlobj.workbooks(1).worksheets(1).range(c).value =
record.getfieldvalue(2)
count = count+1
End Sub

```

索引

B

- BaseView
 - MetaView への追加 65
 - Microsoft Repository の使用 32
 - Microsoft Repository へのエクスポート 167
 - Oracle データベース・タイプの使用 29
 - Snap 用 34
 - 移動 35, 36
 - 記述 26
 - 結合グループ 48
 - 結合グループの定義 48
 - 結合の更新 57
 - 結合の削除 45
 - 結合の追加 44
 - 結合の名前の変更 44
 - コピー 37
 - 再同期化 52
 - 作成 27
 - 信頼できるセキュリティ 28
 - 選択 35
 - 定義 6
 - データ・ソースの追加 27
 - 表の再同期化 55
 - 表の再配置 36
 - 表の削除 43, 54
 - 表の追加 42
 - 表の物理名と表示名 41
 - 表のプロパティ 39
 - 表のプロパティの編集 40
 - プロパティの変更 38
 - 変更 38
 - 編集 35
 - 列のプロパティの編集 41
 - 連結結合 47
- BaseView エディタ 35
- BaseView における結合の名前の変更 44
- BaseViews のエクスポート 167
- BaseView の再同期化 52
- between 結合 46
- Bin
 - 表示領域 15

C

- Category 16
 - 削除 67
 - 追加 68
 - 定義 60
 - 名前の変更 67
- Column Chooser 85

D

- Data Flow Service, 切替え 21
- Design Studio
 - 起動 10
 - 閉じる 24
 - ユーザー・オプション 20
- Design Studio の起動 10
- Design Studio の終了 24
- Design Studio を閉じる 24
- DTS 変換のインポート 169

E

- Enterprise Link 3

G

- Grid Transform 85
- Grid, 削除 19

M

- MetaView
 - BaseView の追加 60, 65
 - Category 60
 - Category の追加 68
 - Part 60
 - Part および Category の名前の変更 67
 - Part の制約の追加 71
 - Part の追加 62, 66
 - Part のプロパティ 68
 - アイテムの再配置 68
 - アイテムの削除 67

- 記述 60
- 計算フィールドの追加 73
- コピー 64
- 削除 63
- 作成 60
- 自動作成 30
- 選択 63
- 定義 6
- 名前の変更 65
- 編集 64
- 保存 63
- MetaView のコピー 64
- MetaView の自動作成 30
- MetaView の選択 63
- Microsoft Repository
 - BaseViews のエクスポート 167
 - BaseView の作成 32
 - DTS-Scripting Transform の使用 171
 - DTS 変換のインポート 169
 - オブジェクト階層 169
- Microsoft Repository への BaseView のエクスポート 167

O

- OLE
 - ColumnDescriptor API 183
 - CreateObject メソッド 185
 - EndData メソッド 186
 - RecordObject API 181
 - イベント・プロシージャ 180
 - サーバーの作成 186
 - 自動化 178
- OLE DB
 - OLEDB Source Transform の使用 165
 - プロバイダとしての Enterprise Link の使用 164
- OLTP データベース 2

P

- Part
 - 削除 67
 - 制約 71
 - 追加 62
 - 定義 16, 60
 - 名前の変更 67
 - プロパティ 68

Q

- Query Transform 82

S

- 「Search」 ボタン 15
- Snap
 - 概要 17
 - 定義 16
- Snap BaseView 34
- Snap BaseView の作成 34
- SQL Query Transform 82
- SQL Viewer 83
- SQL エディタ 84
- SQL 問合せ
 - 表示 83
 - 編集 84
 - 列の削除 82
- StarView 144
 - 削除 156
 - 定義 134
 - 変更 145-159
- StarView エディタ 133

T

- Transform
 - DTS-Scripting 171
 - Grid 85
 - OLEDB Source 165
 - SQL Query 82
 - VBScript 176
 - インポート 103
 - エクスポート 103
 - 機能説明 79
 - 定義 5, 16
 - 名前の変更 87
 - 入力と出力 101
 - プランへの追加 85
 - 保存 88
- Transform のインポート 103
- Transform のエクスポート 103
- 「Transform」 メニュー 87

V

- VBScript Transform
 - ColumnDescriptor Object メソッド 183
 - CreateObject メソッド 185
 - EndData メソッド 186
 - OLE サーバーの作成 186
 - RecordObject method 181
 - VBScriptCopy 176
 - VBScriptInplace 176
 - VBScriptSink 177
 - VBScriptSource 176
 - イベント・プロシージャ 180
 - エラー処理 187
 - コンパイル 178

- 送信 178
- 定義 176
- データ型 187
- 編集 177
- 保存 178
- メソッドとイベント 178
- 例 189

- VBScript のコンパイル 178
- Visual Basic Script Transform 176

あ

- アーキテクチャ 8

い

- 移動

- BaseView 35
- データ・フロー 79
- プランのステップ 79

え

- エラー

- VBScript コンパイル 187
- VBScript ランタイム 188

お

- オプション

- BaseView エディタのプロンプト 22
- Bin のプロンプト 23
- Design Studio の設定 20
- 表示 20

か

- 階層, 属性 154
- 外部結合 45
- カスタム・プラン・プロパティ 109
 - 型 111
 - 計算 Part 74
 - 削除 111
 - 追加 109

き

- キー・フィールド・グループ
 - サンプル・データ・セット 100
 - 反復 95
- 業務系データベース 2

く

- クエリー・エディタ 83

け

- 計算 Part

- 演算子 74
- 関数 74
- 作成 75

- 計算フィールド, 定義 73

- 結合

- between 46
- 外部 45
- グループ 48
- 再同期化 57
- 削除 45
- 追加 44
- 内部 45
- 名前の変更 44
- 連結 47

- 結合タイプ 45

こ

- ごみ箱, プランの移動 19

さ

- 削除

- Grid 19
- Grid からの列の削除 85
- MetaView 63
- 結合 45
- 結合グループ 49
- データ・フロー・ステップ 87
- データ・フロー接続 86

- 作成

- BaseView 27
- MetaView 60
- VBScript Transform 178
- 結合グループ 48
- データ・フロー 85
- プラン 78
- プラン・プロンプト 118
- 別名表 50

- サブプラン

- 移動 93
- キー・フィールド・グループによる反復 95
- ステップのグループ化 92
- ステップのグループ解除 93
- 接続点 94
- 追加 91
- データ・フローでの使用 93
- データ・フローの作成 94
- 反復の有効化 94
- 反復プロパティの設定 96
- 複数のデータ・フロー 96
- レコードによる反復 95

- サブプランの移動 93

サブプランのステップのグループ化 92
サブプランのステップのグループ解除 93

し

自己結合 51

集計

StarView を参照

統計のリフレッシュ 159

ナビゲート 132

論理ディメンション 144

論理ファクト 137

集計階層, 編集 144

集計関数のデフォルト 148

集計表の移入 143

書式設定ツールバー, 定義 15

信頼できるセキュリティ 28

す

ステージング領域, 定義 15

ステータス・バー 15

ステップ 86

削除 87

名前の変更 87

プロパティ 89

編集 90

ステップのプロパティの表示 90

せ

制約 71

接続設定, 変更 11

接続点, サブプラン 94

そ

属性階層

複数パスの作成 155

編集 154

た

タブ 15

つ

追加

BaseView 27

サブプラン 91

表の BaseView への追加 42

プランへのカスタム・スクリプト 173

ツールバー

書式設定 15

データ・ツール 15

標準 14

て

ディメンション表, 集計の作成 142

ディメンション, 論理 144

データ・ウェアハウス 2

データ・ツール・ツールバー 15

データの移動 5

データ・フロー・エディタ 78

Transform の追加 85

ステップ 86

ステップの編集 90

開く 78

データ・フローでのズーム操作 80

データ・フローの印刷 80

データ・フロー・ハンドル, 定義 15

データ・フロー・プラン

SQL Query ソース 82

印刷 80

作成 85

シンク 86

ステップの削除 87

ステップの名前の変更 87

接続の削除 86

ソース 86

定義 5

ビューの拡大 80

ビューの縮小 80

データ・マート 2,5

と

問合せ

表示 83

編集 84

列の削除 82

統計, 集計 159

な

内部結合 45

は

パスワード, 変更 13

ハンドル

データ・フロー 15

プラン 16

反復

キー・フィールド・グループ 95

サブプランでの有効化 94

設定のプロパティ 96

データ・セットの例 99

非反復データ・フロー 96

レコード 95

ひ

表

BaseView からの削除 43

BaseView での再同期化 55

結合の追加 44

追加 42

物理名と表示名 41

標準ツールバー 14

表のプロパティ

表示 39

編集 40

ふ

ファクト・コンテナ 137

ファクト表, 集計の作成 141

フィルタ, カスタム・プラン・プロパティの使用 116

プラン

概要 17

共有 7

ごみ箱への移動 19

定義 5, 16

データ・フロー・エディタ 78

保存 18

プラン・ツリー 79

プラン・ハンドル 16

プラン・プロパティ

型 111

削除 111

追加 109

表示 109

フィルタ 116

プラン・プロンプト

作成 118

設定 123

プレビュー 124

変更 123

プリファレンス, 「ユーザー・オプション」を参照

プログラミング機能 7

プログレス・バー 15

プロパティ

Part 68

カスタムの追加 109

カスタムの表示 109

型 111

削除 111

表 39

プロパティ・ビューア 109

へ

別名表

作成 50

自己結合での使用 51

変更

BaseView のプロパティ 38

結合グループ 49

接続設定 11

プラン・プロンプト 123

編集

MetaView 64

VBScript 177

ステップのプロパティ 90

属性階層 154

データ・フロー・プラン 78

ほ

保存

MetaView 63

Transform 88

VBScript Transform 178

ゆ

ユーザー・オプション

BaseView エディタのプロンプト 22

Bin のプロンプト 23

Design Studio の設定 20

表示 20

ユーザー・オプションの変更 20

れ

例

VBScriptCopy 190

VBScriptInplace 189

VBScriptSink 192

VBScriptSource 191

列のプロパティ

表示 40

編集 41

連結結合 47

ろ

論理ディメンション 144

論理ファクト 137

わ

ワークスペース

消去 19

定義 15

ワークスペースの消去 19

