



## インフォメーションデザインツールユーザガイド

■ SAP BusinessObjects Business Intelligence platform 4.0 Support Package 2

2011-04-12

## 著作権

© 2011 SAP AG. All rights reserved. SAP、R/3、SAP NetWeaver、Duet、PartnerEdge、ByDesign、SAP Business ByDesign、および本書に記載されたその他のSAP製品、サービス、ならびにそれぞれのロゴは、ドイツおよびその他の国々におけるSAP AGの商標または登録商標です。Business ObjectsおよびBusiness Objectsロゴ、BusinessObjects、Crystal Reports、Crystal Decisions、Web Intelligence、Xcelsius、および本書で引用されているその他のBusiness Objects製品、サービス、ならびにそれぞれのロゴは、米国およびその他の国々におけるBusiness Objects S.A.の商標または登録商標です。Business ObjectsはSAPのグループ企業です。本書に記載されたその他すべての製品およびサービス名は、それぞれの企業の商標です。本書に記載されたデータは情報提供のみを目的として提供されています。製品仕様は、国ごとに変わる場合があります。これらの文書の内容は、予告なしに変更されることがあります。また、これらの文書はSAP AGおよびその関連会社(「SAPグループ」)が情報提供のためにのみ提供するもので、いかなる種類の表明および保証を伴うものではなく、SAPグループは文書に関する誤記・脱落等の過失に対する責任を負うものではありません。SAPグループの製品およびサービスに対する唯一の保証は、当該製品およびサービスに伴う明示的保証がある場合に、これに規定されたものに限られます。本書のいかなる記述も、追加の保証となるものではありません。

2011-04-12

# 目次

第 1 章	ドキュメント履歴.....	13
第 2 章	インフォメーションデザインツールについて.....	15
2.1	インフォメーションデザインツールの概要.....	15
2.1.1	インフォメーションデザインツールとは.....	15
2.1.2	インフォメーションデザインツールの使用者について.....	16
2.2	リソースおよびワークフローの概要.....	16
2.2.1	インフォメーションデザインツールで使用可能なデザインリソースとは.....	17
2.2.2	インフォメーションデザインツールを使用したユニバースの作成方法.....	18
2.2.3	インフォメーションデザインツールのユーザ基本設定.....	21
2.2.4	ユーザインタフェースの表示のリセット.....	21
第 3 章	プロジェクトの使用.....	23
3.1	ローカルプロジェクトについて.....	23
3.1.1	ローカルプロジェクトの作成.....	24
3.1.2	リソース名.....	24
3.1.3	レポートとしてのリソースの保存.....	24
3.2	共有プロジェクトについて.....	25
3.2.1	ローカルプロジェクトからの共有プロジェクトの作成.....	26
3.2.2	共有プロジェクトの名前の変更.....	26
3.2.3	共有プロジェクトの削除.....	27
3.3	プロジェクト同期について.....	27
3.3.1	[プロジェクト同期] ビューを開く.....	30
3.3.2	プロジェクトの同期.....	31
3.3.3	リソースのロック.....	32
3.3.4	リソースのロック解除.....	32
3.3.5	共有リソースへの変更の結合.....	33
3.4	リソースの依存関係について.....	34
3.4.1	ローカル依存関係の表示.....	34
3.4.2	リポジトリ依存関係の表示.....	35
3.5	公開されたユニバースの取得.....	35

第 4 章	.unv ユニバースの変換	37
4.1	.unv ユニバースおよび .unx ユニバース	37
4.2	.unv ユニバースの変換について	37
4.3	.unv ユニバースの変換時にサポートされている機能	39
4.4	.unv ユニバース変換後の整合性チェックのエラーの解決方法	44
4.5	リポジトリでの .unv ユニバースの変換	45
4.6	ローカルに保存された .unv ユニバースの変換	46
第 5 章	接続の使用	49
5.1	接続について	49
5.1.1	ローカル接続について	50
5.1.2	セキュリティ接続について	51
5.1.3	接続ショートカットについて	51
5.1.4	SAP NetWeaver BW 接続について	52
5.2	リレーショナル接続の作成	53
5.2.1	接続の名前	53
5.2.2	データソースパラメータの設定	54
5.2.3	ミドルウェアドライバの選択	62
5.3	OLAP 接続の作成	62
5.3.1	OLAP ミドルウェアドライバの選択	63
5.3.2	OLAP データソースのログインパラメータの設定	63
5.3.3	OLAP キューブの選択	67
5.4	接続ショートカットの作成	67
5.5	接続および接続ショートカットの編集について	67
第 6 章	データファンデーションを操作する	69
6.1	データファンデーションの概要	69
6.1.1	データファンデーションの種類について	69
6.2	データファンデーションエディタについて	70
6.3	データファンデーションの作成方法	71
6.4	シングルソースのデータファンデーションについて	73
6.5	複数のソースが有効化されたデータファンデーションについて	73
6.5.1	複数の接続があるデータファンデーション	74
6.5.2	複数のソースが有効化されたデータファンデーションの SQL 式	75
6.5.3	NetWeaver BW 接続を使ったデータファンデーション	75
6.6	データファンデーションの接続について	76
6.6.1	データファンデーションに対する接続の追加	77
6.6.2	データファンデーション内での接続の変更	77
6.7	データファンデーションのテーブルについて	78

6.7.1	データファンデーションへのテーブルの挿入.....	79
6.7.2	データファンデーションテーブルの編集.....	80
6.7.3	テーブル名の太文字/小文字の設定.....	80
6.8	テーブルの値の表示および分析。.....	81
6.9	テーブルキーについて.....	81
6.10	表の行数について.....	82
6.11	列フィルタについて.....	83
6.12	計算列について.....	84
6.13	結合について.....	84
6.13.1	結合の検出について.....	86
6.14	カーディナリティについて.....	87
6.15	派生テーブルについて.....	88
6.16	テーブルの結合について.....	89
6.17	エイリアステーブルについて.....	90
6.18	コンテキストについて.....	90
6.19	ループの解決.....	91
6.20	入力列について.....	92
6.21	データファンデーション内のパラメータと値の一覧について.....	93
6.22	データファンデーションのプロパティについて.....	93
6.23	ローカル依存関係の表示.....	94
6.24	データファンデーションの最新表示について.....	94
6.25	カスタムデータファンデーションビューについて.....	95
6.26	テーブルファミリについて.....	95
6.27	データファンデーションビューでのコメントの挿入.....	96
6.28	データファンデーションでのテーブルおよび列の検索.....	96
6.29	選択時のビューの中央揃え.....	97
<b>第 7 章</b>	<b>ビジネスレイヤの使用.....</b>	<b>99</b>
7.1	ビジネスレイヤとは.....	99
7.2	ビジネスレイヤの作成.....	99
7.2.1	ビジネスレイヤの新規作成ウィザードの使い方.....	100
7.3	ビジネスレイヤエディタについて.....	103
7.4	ビジネスレイヤのプロパティについて.....	104
7.4.1	OLAP データソースプロパティ.....	105
7.4.2	ビジネスレイヤのプロパティの編集.....	106
7.4.3	ビジネスレイヤのデータソースの変更.....	107
7.5	ビジネスレイヤオブジェクトの使用.....	107
7.5.1	ビジネスレイヤオブジェクトについて.....	107
7.5.2	ビジネスレイヤへのオブジェクトの挿入.....	119
7.5.3	データファンデーションからのオブジェクトの直接挿入.....	119
7.5.4	ビジネスレイヤオブジェクトの編集.....	120

7.5.5	ビジネスレイヤオブジェクトの検索.....	120
7.5.6	ビジネスレイヤのツリービューの表示オプションの変更.....	121
7.5.7	オブジェクトのキーの定義.....	121
7.6	ビジネスレイヤビューについて.....	122
7.6.1	ビジネスレイヤビューの作成と編集.....	122
7.6.2	ビジネスレイヤビューによるフィルタの適用.....	123
7.7	ビジネスレイヤのクエリオブジェクトについて.....	123
7.7.1	ビジネスレイヤへのクエリオブジェクトの追加.....	123
7.8	パラメータについて .....	124
7.8.1	パラメータの挿入と編集.....	125
7.9	値の一覧について .....	126
7.9.1	値の一覧のクエリオプション .....	126
7.9.2	値の一覧の列のプロパティ.....	127
7.9.3	値の一覧の挿入および編集 .....	128
7.9.4	値の一覧のビジネスオブジェクトへの関連付け.....	130
7.9.5	値の一覧のビジネスレイヤで定義済みのプロンプトへの関連付け.....	130
7.10	オブジェクトのナビゲーションパスについて.....	131
7.10.1	ビジネスレイヤへのナビゲーションパスオブジェクトの挿入.....	132
7.11	アグリゲート認識について.....	132
7.11.1	アグリゲート認識の設定.....	133
7.12	ビジネスレイヤの参照について.....	133
7.12.1	OLAP キューブに基づくビジネスレイヤの最新表示 .....	134
7.13	最適化されたクエリ実行のための統計の計算について .....	134
7.13.1	複数ソースユニバースの統計の計算.....	135
<b>第 8 章</b>	<b>クエリパネルの使用法.....</b>	<b>137</b>
8.1	クエリパネルについて.....	137
8.1.1	クエリパネルの説明.....	137
8.1.2	[結果オブジェクト] 枠について .....	139
8.1.3	[クエリフィルタ] 枠について.....	139
8.1.4	[データプレビュー] 枠について .....	140
8.1.5	メンバーセレクトについて.....	141
8.2	クエリについて.....	148
8.2.1	クエリのプロパティについて.....	148
8.2.2	作成可能なクエリのタイプについて.....	150
8.2.3	プロンプトの使用.....	153
<b>第 9 章</b>	<b>整合性のチェック.....</b>	<b>155</b>
9.1	[整合性のチェック] について.....	155
9.2	整合性のチェックの実行.....	155
9.3	整合性エラーのチェックのレビュー.....	156

9.4	バックグラウンドでの整合性のチェックの設定.....	156
9.5	ルールの整合性のチェックについて.....	157
9.5.1	複数ソースユニバースのルールの整合性のチェック.....	157
9.5.2	データファンデーションのルールの整合性のチェック .....	158
9.5.3	値の一覧 (LOV) のルールの整合性のチェック.....	159
9.5.4	パラメータのルールの整合性のチェック .....	161
9.5.5	データアクセスセキュリティプロファイルのルールの整合性のチェック.....	162
9.5.6	OLAP ユニバースのルールの整合性のチェック .....	163
9.5.7	OLAP ユニバース内の MDX オブジェクトのルールの整合性のチェック.....	165
<b>第 10 章</b>	<b>リソースの公開.....</b>	<b>169</b>
10.1	リソースの公開について.....	169
10.2	ユニバースの公開.....	170
10.2.1	リポジトリフォルダの選択.....	171
10.2.2	ローカルフォルダの選択.....	171
10.3	ローカル接続のリポジトリへの公開.....	171
<b>第 11 章</b>	<b>リポジトリリソースの使用.....</b>	<b>173</b>
11.1	リポジトリのリソースの管理について.....	173
11.2	セッション管理について.....	174
11.2.1	セッションを開く.....	175
11.2.2	セッションの終了.....	176
11.3	リポジトリ内で公開されたユニバースでのクエリの実行.....	176
<b>第 12 章</b>	<b>セキュリティの管理.....</b>	<b>177</b>
12.1	インフォメーションデザインツールのセキュリティの概要.....	177
12.1.1	ユニバースセキュリティの概要.....	177
12.1.2	インフォメーションデザインツールでのリソースのセキュリティについて.....	179
12.1.3	インフォメーションデザインツール ユーザの CMC 権限.....	179
12.2	データセキュリティプロファイル.....	182
12.2.1	データセキュリティプロファイルの接続設定.....	183
12.2.2	データセキュリティプロファイルの制御設定.....	183
12.2.3	データセキュリティプロファイルの SQL 設定.....	184
12.2.4	データセキュリティプロファイルの行設定.....	185
12.2.5	データセキュリティプロファイルのテーブル設定.....	186
12.3	ビジネスセキュリティプロファイル.....	186
12.3.1	ビジネスセキュリティプロファイルのクエリの実行設定.....	187
12.3.2	ビジネスセキュリティプロファイルの表示データ設定.....	189
12.3.3	ビジネスセキュリティプロファイルのフィルタ設定.....	190
12.4	セキュリティプロファイルの集計.....	191

12.4.1	接続設定の集計.....	192
12.4.2	制御設定の集計.....	192
12.4.3	SQL 設定の集計.....	193
12.4.4	行設定の集計.....	194
12.4.5	テーブル設定の集計.....	195
12.4.6	クエリの作成設定の集計.....	195
12.4.7	表示データ設定の集計.....	197
12.4.8	フィルタ設定の集計.....	198
12.5	セキュリティエディタの使用.....	199
12.5.1	セキュリティエディタを開く.....	200
12.5.2	セキュリティプロファイルの挿入と編集.....	200
12.5.3	データセキュリティプロファイルの優先順位の変更.....	201
12.5.4	セキュリティプロファイルの集計オプションの変更.....	201
12.5.5	ユーザへのセキュリティプロファイルの割り当て.....	202
12.5.6	ユーザに割り当てられているプロファイルの表示とネットプロファイルのプレビュー.....	202
<b>第 13 章</b>	<b>関数のリファレンス.....</b>	<b>205</b>
<b>第 14 章</b>	<b>SQL 生成パラメータのリファレンス.....</b>	<b>209</b>
14.1	SQL 生成パラメータについて.....	209
14.2	SQL 生成パラメータのリファレンス.....	209
14.2.1	ANSI92.....	209
14.2.2	AUTO_UPDATE_QUERY .....	210
14.2.3	BEGIN_SQL.....	210
14.2.4	BLOB_COMPARISON.....	211
14.2.5	BOUNDARY_WEIGHT_TABLE .....	212
14.2.6	COLUMNS_SORT .....	213
14.2.7	CUMULATIVE_OBJECT_WHERE .....	213
14.2.8	DISABLE_ARRAY_FETCH_SIZE_OPTIMIZATION .....	214
14.2.9	DISTINCT_VALUES .....	215
14.2.10	END_SQL .....	215
14.2.11	EVAL_WITHOUT_PARENTHESES .....	216
14.2.12	FORCE_SORTED_LOV .....	217
14.2.13	INNERJOIN_IN_WHERE .....	217
14.2.14	JOIN_BY_SQL .....	218
14.2.15	MAX_INLIST_VALUES .....	218
14.2.16	REPLACE_COMMA_BY_CONCAT .....	219
14.2.17	SELFJOINS_IN_WHERE .....	220
14.2.18	SHORTCUT_BEHAVIOR.....	220
14.2.19	SMART_AGGREGATE .....	221
14.2.20	THOROUGH_PARSE .....	222



14.2.21	TRUST_CARDINALITIES .....	222
14.2.22	UNICODE_STRINGS.....	223
<b>第 15 章</b>	<b>SQL 関数リファレンス.....</b>	<b>225</b>
15.1	集計関数.....	225
15.1.1	AVG.....	225
15.1.2	COUNT.....	226
15.1.3	MAX.....	226
15.1.4	MIN.....	227
15.1.5	SUM.....	228
15.2	数値関数.....	228
15.2.1	abs.....	228
15.2.2	acos.....	229
15.2.3	asin.....	229
15.2.4	atan.....	229
15.2.5	atan2.....	229
15.2.6	ceiling.....	230
15.2.7	cos.....	230
15.2.8	cot.....	230
15.2.9	degrees.....	231
15.2.10	exp.....	231
15.2.11	floor.....	231
15.2.12	log.....	232
15.2.13	log10.....	232
15.2.14	mod.....	232
15.2.15	pi.....	233
15.2.16	power.....	233
15.2.17	radians.....	233
15.2.18	rand.....	234
15.2.19	round.....	234
15.2.20	sign.....	235
15.2.21	sin.....	235
15.2.22	sqrt.....	235
15.2.23	tan.....	236
15.2.24	trunc.....	236
15.3	日付/時刻関数.....	237
15.3.1	curdate.....	237
15.3.2	curtime.....	237
15.3.3	dayName.....	237
15.3.4	dayNameL.....	238
15.3.5	dayOfMonth.....	239

15.3.6	dayOfWeek.....	239
15.3.7	dayOfWeekL.....	239
15.3.8	dayOfYear.....	240
15.3.9	decrementDays.....	240
15.3.10	hour.....	241
15.3.11	incrementDays.....	241
15.3.12	minute.....	241
15.3.13	month.....	242
15.3.14	monthName.....	242
15.3.15	monthNameL.....	243
15.3.16	now.....	243
15.3.17	quarter.....	244
15.3.18	second.....	244
15.3.19	timestampadd.....	244
15.3.20	timestampdiff.....	245
15.3.21	trunc.....	246
15.3.22	week.....	247
15.3.23	weekL.....	247
15.3.24	year.....	248
15.4	文字列関数.....	248
15.4.1	ascii.....	249
15.4.2	char.....	249
15.4.3	concat.....	249
15.4.4	containsOnlyDigits.....	250
15.4.5	insert.....	250
15.4.6	isLike.....	251
15.4.7	left.....	252
15.4.8	leftStr.....	252
15.4.9	len.....	253
15.4.10	IPad.....	253
15.4.11	lTrim.....	254
15.4.12	permute.....	254
15.4.13	pos.....	256
15.4.14	repeat.....	257
15.4.15	replace.....	257
15.4.16	replaceStringExp.....	258
15.4.17	right.....	258
15.4.18	rightStr.....	259
15.4.19	rPad.....	259
15.4.20	rPos.....	259
15.4.21	rTrim.....	260

15.4.22	space.....	261
15.4.23	subString.....	261
15.4.24	toLowerCase.....	262
15.4.25	toUpperCase.....	262
15.4.26	trim.....	262
15.5	システム関数.....	263
15.5.1	データベース.....	263
15.5.2	ifElse.....	263
15.5.3	nvl.....	264
15.5.4	user.....	265
15.6	変換関数.....	265
15.6.1	cast.....	266
15.6.2	convert.....	266
15.6.3	hexaToInt.....	267
15.6.4	intToHexa.....	267
15.6.5	toBoolean.....	268
15.6.6	toBooleanL.....	268
15.6.7	toDate.....	269
15.6.8	toDecimal.....	270
15.6.9	toDecimalL.....	271
15.6.10	toDouble.....	272
15.6.11	toDoubleL.....	272
15.6.12	toInteger.....	273
15.6.13	toIntegerL.....	274
15.6.14	toNull.....	275
15.6.15	toString.....	276
15.6.16	toStringL.....	277
15.6.17	toTime.....	279
15.6.18	toTimeL.....	279
15.6.19	toTimestamp.....	280
15.6.20	val.....	281
15.7	複数のソースを使用するデータファンデーションの関数のためのロケールコード.....	282
付録 A	より詳しい情報.....	283
	索引.....	285



## ドキュメント履歴

以下の表は、最も重要なドキュメント変更の概要です。

バージョン	日付	説明
SAP BusinessObjects Business Intelligence Suite 4.0 サポートパッケージ 2	2011 年 3 月 4 日	<p>大文字/小文字の設定コマンド (データファンデーションエディタ) について追加されたドキュメンテーションは、80 ページの「<a href="#">テーブル名の大文字/小文字の設定</a>」を参照してください。</p> <p>選択部分のセンタリングコマンド (データファンデーションエディタ) について追加されたドキュメンテーションは、97 ページの「<a href="#">選択時のビューの中央揃え</a>」を参照してください。</p> <p>選択したメンバーのメンバーセクタへの表示について追加されたドキュメンテーションは、148 ページの「<a href="#">選択したメンバーのメンバーセクタへの表示</a>」を参照してください。</p> <p>クエリおよびレポーティングアプリケーションのユーザが、SAP NetWeaver BW で複数のソースが有効化されたユニバースにアクセスするのに必要な認証について追加されたリンクは、52 ページの「<a href="#">SAP NetWeaver BW 接続について</a>」を参照してください。</p>



# インフォメーションデザインツールについて

## 2.1 インフォメーションデザインツールの概要

### 2.1.1 インフォメーションデザインツールとは

インフォメーションデザインツールは、SAP BusinessObjects のメタデータデザイン環境で、デザイナーが SAP BusinessObjects ユニバースを作成およびデプロイするために、リレーショナルおよび OLAP ソースからメタデータを抽出、定義、および編集できるようにします。

ユニバースとは、編成されたメタデータオブジェクトのコレクションのことで、これにより、専門用語を使わずに、ビジネスユーザが企業のデータを分析してレポートを作成できます。これらのオブジェクトには、ディメンション、メジャー、階層、属性、定義済みの計算、関数、およびクエリが含まれます。メタデータオブジェクトレイヤは、ビジネスレイヤといい、リレーショナルデータベースのスキーマまたは OLAP キューブ上で構築されるため、オブジェクトは SQL 式または MDX 式を介して直接データベース構造にマップされます。ユニバースには、データソースを識別する接続が含まれるため、クエリをデータ上で実行できます。

ユニバースの役割は、ビジネスユーザに意味を理解できるビジネスオブジェクトを提供することです。ユーザは、基になるデータソースと構成にかかわらず、関連するビジネス用語を使用して、自由にデータを分析してレポートを作成できます。

インフォメーションデザインツールを使用して作成されたユニバースは、次の BI 4 で始まるバージョンの SAP BusinessObjects のデータ分析アプリケーションおよびレポートングアプリケーションで使用できます。

- ・ SAP BusinessObjects Web Intelligence
- ・ SAP Crystal Reports for Enterprise
- ・ SAP BusinessObjects Explorer
- ・ SAP BusinessObjects Dashboard Design

デザイナーを有効化してユニバースを作成するため、インフォメーションデザインツールは次のことに必要なリソースを提供します。

- ・ データソースへの接続を作成する。
- ・ 完全な OLAP キューブスキーマを抽出する。
- ・ データファンデーションと呼ばれるリレーショナルスキーマを作成するために、テーブルおよび結合を抽出する。

- ・ キューブまたはデータファンデーションからメタデータオブジェクトを作成する。これらのオブジェクトは、ビジネスレイヤに含まれ編成される。オブジェクト内の SQL および MDX 式は検証でき、ビジネスレイヤをテストするため、クエリはターゲットデータベースに対して実行される。
- ・ 複数のデザイナーが同じリソースで同時に作業できるように、リソースを共有する。
- ・ ビジネスレイヤ、データファンデーションおよび単一のユニバースファイル (.unx) への接続をコンパイルするユニバースを公開する。
  - ・ SAP BusinessObjects データ分析およびレポーティングアプリケーションのデプロイメントで実装されるように、ユニバースをリポジトリに公開する。
  - ・ スタンドアロンモードのクライアントアプリケーション（たとえば、Web Intelligence Rich Client）で実装されるように、ローカルにユニバースを公開します。
- ・ ユニバースデータおよびメタデータへのユーザアクセスを定義するセキュリティプロファイルを作成する。

#### 関連項目

- ・ 17 ページの [インフォメーションデザインツールで使用可能なデザインリソースとは](#)

## 2.1.2 インフォメーションデザインツールの使用者について

インフォメーションデザインツールは、メタデータモデリングアプリケーションです。SAP BusinessObjects ユニバースを作成してデプロイするために、メタデータの抽出、デザイン、リソースの公開機能を提供します。インフォメーションデザインツールのユーザプロファイルは、旧バージョンの BusinessObjects Universe Designer を使用するユニバースデザイナーと同じです。

一般的に、データベース管理、アプリケーション管理またはアプリケーション開発、プロジェクト管理などの担当者、または、他のユーザのためにユニバースを作成するのに必要なスキルを身に付けたレポート作成者がユニバースデザイナーとなります。セキュリティ管理者も、インフォメーションデザインツールを使用してユニバースセキュリティプロファイルを定義します。

社内に複数のユニバースデザイナーがいることは珍しくありません。必要なデザイナーの人数はそれぞれの会社のデータニーズに応じて異なります。たとえば、デザイナーをアプリケーション、プロジェクト、部門、業務範囲別に1人ずつ配置することもできます。

複数のデザイナーがユニバースを作成する場合は、用語に関するルールセットまたはガイドラインを定義して、オブジェクトの表現に整合性が保たれるようにすることが重要です。

## 2.2 リソースおよびワークフローの概要



## 2.2.1 インフォメーションデザインツールで使用可能なデザインリソースとは

インフォメーションデザインツールは、次のようなデザインリソースを提供します。これらのリソースは、デザイナーが、リレーショナルデータソースまたは OLAP データソースからメタデータを最初に抽出し、次に、特定のユーザグループを対象にするオブジェクトのビジネスレイヤを作成するのに使用します。

リソース	説明
プロジェクト	<p>プロジェクトは、名前の付いたローカルワークスペースです。プロジェクトには、1 つまたは複数のユニバースを作成するのに使用される 1 つまたは複数のリソースが含まれます。</p> <p>複数のデザイナーが同じリソースで作業できるようにするため、プロジェクトを共有することができます。</p>
接続	<p>接続は、名前の付いたパラメータの集合のことで、リレーショナルまたは OLAP データベースに対するユニバースのアクセス方法を定義します。</p> <p>接続はローカルファイルにできるほか、インフォメーションデザインツールでローカルショートカットから参照される共有リポジトリ内のオブジェクトにすることもできます。ユニバースは通常、少なくとも 1 つの接続と関連付けられます。</p>
データファンデーション	<p>データファンデーションはスキーマのことで、1 つまたは複数のリレーショナルデータベースから関連するテーブルおよび結合を定義します。デザイナーは、コンテキスト、プロンプト、計算列、およびその他の SQL 定義によってデータファンデーションを拡張します。データファンデーションは、1 つまたは複数のビジネスレイヤの基本になります。</p>
ビジネスレイヤ	<p>ビジネスレイヤは、メタデータオブジェクトのコレクションのことで、リレーショナルデータベースエンティティまたは OLAP キューブの抽象化を提供し、ビジネスユーザに分かりやすいものです。オブジェクトは、SQL 式を介して基になるデータファンデーション、または、MDX 式を介して基になる OLAP キューブにマップされます。これらのオブジェクトには、ディメンション、階層、メジャー、属性、および定義済みの条件が含まれます。</p> <p>ビジネスレイヤは、作成中のユニバースであり、ビジネスレイヤが完成すると、ユニバースとして、接続または接続ショートカット、およびデータファンデーションを使用してコンパイル、公開、デプロイされます。</p>
クエリ	<p>クエリは、オブジェクトのセットのことで、データベースに対するデータの要求を定義します。クエリは、ビジネスレイヤでオブジェクトをテストするのに使用されるビジネスレイヤに、メタデータオブジェクトとして定義または保存することができます。</p>

リソース	説明
パラメータおよび値の一覧	<p>パラメータは、クエリの実行時に値を必要とするユニバース内の変数です。パラメータは、多くの場合、ユーザに値の指定を促すために定義されており、この場合、プロンプトとして参照されます。</p> <p>値の一覧は、ユニバース内のオブジェクトに関連付けられるデータ値の集合で、ユーザがプロンプトの値を選択できるようにします。</p> <p>パラメータおよび値の一覧は、データファンデーション内で定義できます。これらは、データファンデーションに基づくすべてのビジネスレイヤによって継承されます。</p> <p>パラメータおよび値の一覧は、ビジネスレイヤ内でも定義できます。</p>
ユニバース	<p>ユニバースは、コンパイルされたファイルのことで、ビジネスレイヤのデザインで作成される、メタデータオブジェクトの定義で使用するすべてのリソースを含みます。</p> <p>ユニバースは、SAP BusinessObjects のデータ分析および、分析とレポートング用にビジネスレイヤオブジェクトが表示されるレポートングアプリケーションで使用されます。</p>

#### 関連項目

- ・ 24 ページの[ローカルプロジェクトの作成](#)」
- ・ 49 ページの[接続について](#)」
- ・ 69 ページの[データファンデーションの概要](#)」
- ・ 99 ページの[ビジネスレイヤとは](#)」
- ・ 123 ページの[ビジネスレイヤのクエリオブジェクトについて](#)」
- ・ 124 ページの[パラメータについて](#)」
- ・ 169 ページの[リソースの公開について](#)」

## 2.2.2 インフォメーションデザインツールを使用したユニバースの作成方法

ビジネスレイヤをリポジトリまたはファイルシステムに単一ファイル(.unx)として公開するときに、ユニバースを作成します。この節では、SAP BusinessObjects ユニバースを作成およびデプロイするために、インフォメーションデザインツール内で必要なリソースを作成および使用するためのプロセス全体について説明します。

2 種類のデータソースからユニバースを作成できます。

- ・ 1 つまたは複数のリレーショナルデータベース
- ・ OLAP キューブ

ユニバースを作成するためのプロセスは、リレーショナルデータソースおよび OLAP データソースの両方ではほぼ同じですが、リレーショナルソースに基づくユニバースの場合は、ビジネスレイヤを作成する前に、データファンデーションを作成する必要があります。これは、ユニバース用に OLAP キューブへの接続を使用している場合ではなく、オブジェクトがキューブ構造から表示されており、ユニバースのオブジェクトを直接選択できる場合です。

ユニバースを作成するには、次のプロセスに従います。

ユニバースタスク	リレーショナルへの適用	OLAP への適用	説明
ローカルプロジェクトの作成	○	○	<p>プロジェクトは、ユニバースの作成に必要なリソースを含むインフォメーションデザインツール内で定義されたワークスペースです。ユニバースを作成する前に、リソースを保存するためのプロジェクトを作成する必要があります。</p> <p>プロジェクトは、たとえば、データファンデーション、ビジネスレイヤ、および接続などの独立したリソースをいくつでも含むことができます。プロジェクトに含まれるすべてのリソースは、交互に使用可能です。たとえば、接続は、同じプロジェクト内で複数のデータファンデーションから使用できます。</p>
接続の定義	○	○	<p>リレーショナルデータベースまたは OLAP キューブへの接続を作成します。[新規接続]ウィザードを使用して、接続パラメータを定義して保存します。</p> <p>接続は、独立リソースです。リレーショナル接続は、複数のデータファンデーションで使用できます。OLAP 接続またはデータファンデーションは、複数のビジネスレイヤで使用できます。接続は、ファイルシステムにローカルファイルとして保存するか、リポジトリ内に保護された接続として保存することができます。保護された接続には、リポジトリ内の保護された接続を参照するために、ローカルプロジェクトに接続ショートカットリソースが必要です。</p> <p>[リポジトリリソース] ビューで、保護された接続をリポジトリ内で直接作成し、接続ショートカットを作成することができます。</p>

ユニバースタスク	リレーショナルへの適用	OLAP への適用	説明
データファンデーションの作成	○	×	<p>リレーショナルデータベースに基づくユニバースのデータファンデーションを作成します。シングルソースのユニバースの場合、データベースソースを識別する単一接続から始めます。</p> <p>公開済みユニバースが複数のデータソースにアクセスできるように、複数の接続でデータファンデーションを作成することもできます。この場合、複数ソース有効データファンデーションを作成します。</p> <p>OLAP 接続に基づくユニバースのデータファンデーションは作成しません。メタデータオブジェクトは、キューブから直接使用できるので、ビジネスレイヤの作成に中間のマッピングレイヤは必要ありません。</p>
ビジネスレイヤの作成	○	○	<p>ビジネスレイヤは、データファンデーションまたは OLAP キューブに基づいて作成するメタデータオブジェクトのコレクションです。ビジネスレイヤの新規作成ウィザードを使用して、データファンデーションまたは OLAP キューブのいずれかのデータソースを選択します。</p> <p>リレーショナルビジネスレイヤの場合、すべてのデータファンデーションのビジネスレイヤで自動的にオブジェクトを作成する方法と、オブジェクトとしてマップされる列を選択する方法を選択できます。</p> <p>OLAP ビジネスレイヤの場合、オブジェクトはキューブ内のすべての構造に対して作成されます。</p> <p>オブジェクトは、[ビジネスレイヤ] ペインに表示されます。ユニバースデザインで必要な場合、ディメンション、階層、メジャー、属性、およびその他のオブジェクトを追加できます。いつでも SQL または MDX を検証することができます。クエリ、値の一覧、パラメータ(プロンプトともいう) およびナビゲーションパスオブジェクトを作成できます。</p> <p>リレーショナルビジネスレイヤを複数のデータソースで作成するには、ビジネスレイヤが複数ソース有効データファンデーションに基づいている必要があります。</p>
ユニバースの公開	○	○	<p>ビジネスレイヤをユニバース(.unx) ファイルとしてリポジトリ、またはローカルファイルシステムに公開します。</p>

ユニバースタスク	リレーショナルへの適用	OLAP への適用	説明
ユニバースセキュリティの定義	○	○	インフォメーションデザインツール でセキュリティエディタを使用してリポジトリに公開されたユニバースで、セキュリティを定義します。

#### 関連項目

- ・ 23 ページの[ローカルプロジェクトについて](#)
- ・ 53 ページの[リレーショナル接続の作成](#)
- ・ 62 ページの[OLAP 接続の作成](#)
- ・ 71 ページの[データファンデーションの作成方法](#)
- ・ 99 ページの[ビジネスレイヤの作成](#)
- ・ 169 ページの[リソースの公開について](#)
- ・ 177 ページの[ユニバースセキュリティの概要](#)
- ・ 173 ページの[リポジトリのリソースの管理について](#)

## 2.2.3 インフォメーションデザインツールのユーザ基本設定

以下を含む、インフォメーションデザインツールの動作や表示設定をカスタマイズして、基本設定を行うことができます。

- ・ ルールの整合性のチェックの設定
- ・ データファンデーションの自動検出ルールの設定
- ・ データファンデーションの画像表示オプションの設定
- ・ ユーザインタフェースの言語の設定

- 1 [ウィンドウ] > [基本設定] を選択します。  
[設定] ダイアログボックスが表示されます。
- 2 アプリケーションの基本設定を行うには、[インフォメーションデザインツール] のノードを展開し、変更する基本設定の種類に移動します。
- 3 基本設定を行ったら、[適用] をクリックしてから [OK] をクリックします。

## 2.2.4 ユーザインタフェースの表示のリセット

インフォメーションデザインツールのユーザインタフェースは、エディタタブとビューのドラッグアンドドロップ、ビューの最小化、ビュー内でのパネルの非表示や分割によって、カスタマイズすることができます。

ユーザインタフェースをデフォルト設定にリセットするには、[ウィンドウ]>[デフォルト表示にリセット]を選択します。

## プロジェクトの使用

### 3.1 ローカルプロジェクトについて

インフォメーションデザインツールでリソースを作成する最初のステップは、[ローカルプロジェクト]ビューでローカルプロジェクトを作成することです。すべてのリソース（セキュリティ接続とセキュリティプロファイル）をローカルプロジェクトで作成および編集します。

ローカルプロジェクトのリソースとフォルダは、ローカルファイルシステムに物理的ファイルおよびフォルダとして保存されます。[ローカルプロジェクト]ビューでは、インフォメーションデザインツールでローカルプロジェクトをナビゲートし、リソースを開くことができます。

ローカルプロジェクトを作成したら、以下の方法でそこにリソースを入れられます。

- ・ [新規]メニューから使用できるウィザードを使用してリソースを作成できます。
- ・ ユニバースデザインツールを使用して作成した、または旧バージョンから移行した .unv ユニバースを変換します。
- ・ 公開されたユニバースの取得

ローカルプロジェクトでリソース名をダブルクリックすると、インフォメーションデザインツールエディタを使ってリソースを編集できます。

ローカルリソースに行えるその他の操作には、次のものがあります。

- ・ 他のデザイナとリソースを共有できるように共有プロジェクトを作成する。
- ・ データファンデーションおよびビジネスレイヤの整合性をチェックする。
- ・ ローカルファイルシステムまたはリポジトリにビジネスレイヤをユニバースとして公開する。
- ・ リポジトリに接続を公開する。
- ・ 依存関係のあるリソースを表示する。
- ・ レポートとしてリソースを保存する。


これらの操作の詳細については、関連項目を参照してください。

#### 関連項目

- ・ 24 ページの[ローカルプロジェクトの作成](#)」
- ・ 37 ページの[.unv ユニバースの変換について](#)」
- ・ 35 ページの[公開されたユニバースの取得](#)」
- ・ 25 ページの[共有プロジェクトについて](#)」
- ・ 155 ページの[\[整合性のチェック\]について](#)」
- ・ 169 ページの[リソースの公開について](#)」
- ・ 34 ページの[リソースの依存関係について](#)」

- ・ 24 ページの[レポートとしてのリソースの保存](#)

### 3.1.1 ローカルプロジェクトの作成

- 1 [ファイル] > [新規] > [プロジェクト] を選択します。
- 2 プロジェクトに一意の名前を指定します。
- 3 [プロジェクトの保存場所] テキストボックスに、デフォルトのローカルファイルのパスが表示されます。ローカルプロジェクトを保存する別のローカルフォルダを選択するには、[プロジェクトの保存場所] テキストボックスの  をクリックします。
- 4 [完了] をクリックします。

プロジェクトは、ローカルファイルシステム内に作成され、[ローカルプロジェクト] ビューに表示されます。

### 3.1.2 リソース名

リソース名は、ローカルプロジェクト内の接続、データファンデーション、およびビジネスレイヤを識別します。リソースの作成時に名前を入力します。名前は、ローカルプロジェクト内で一意である必要があります。


任意で、リソースの説明を入力することができます。

#### 関連項目

- ・ 17 ページの[インフォメーションデザインツールで使用可能なデザインリソースとは](#)

### 3.1.3 レポートとしてのリソースの保存

ローカルプロジェクト内のすべてのリソースは、レポートとしてローカルファイルに保存できます。

- 1 レポートを生成するには、[ローカルプロジェクト] ビューでリソース名を右クリックし、[名前を付けて保存] を選択します。
- 2 [レポートの生成] ダイアログボックスで、レポートのファイルパスとレポート名を入力します。[レポートの場所] テキストボックスで  をクリックします。

ローカルファイルシステム内の場所を選択します。生成されるレポートのファイルの種類には、.pdf、.html、または .txt があります。

- 3 より大きなリソースの場合 (データファンデーションおよびビジネスレイヤ)、[メタデータ要素] ボックスでチェックボックスを選択することにより、レポート内に含めるメタデータ要素を選択できます。



- 4 [生成] をクリックしてレポートを作成します。

## 3.2 共有プロジェクトについて

共有プロジェクトは、他のデザイナーがリソースを使用できるようにするリポジトリ内のプロジェクトです。リソースは、他のデザイナーが作業できるように、ローカルおよび共有プロジェクト間で同期化します。

プロジェクト同期ビューでは、共有プロジェクトとその内容をナビゲートできます。

複数のデザイナーが作業できるリソースを作成する場合は、以下の一般的なワークフローに従います。

- 1 ローカルプロジェクトでリソースを作成します。
- 2 リソースの準備ができたなら、ローカルプロジェクトを共有します。これにより、サーバ上に同じ名前の共有プロジェクトが作成されます。
- 3 プロジェクト内の共有するリソースを同期化します。リソースは、他のデザイナーがアクセスできる共有プロジェクトで使用できます。

他のデザイナーも作業している既存のリソースで作業する場合は、以下の一般的なワークフローに従います。

- 1 プロジェクト同期ビューから、選択した共有プロジェクトで、作業するリソースをロックします。

### 注

ロックは、デザイナー間の通信ツールとして使用できます。他のデザイナーがプロジェクト同期ビューを開くと、変更が加えられていることがロックによって通知されます。また、ロックしている間は、共有プロジェクトで他のデザイナーがリソースを更新できなくなります。ただし、必要に応じてあらゆるデザイナーがロックを外せます。

- 2 サーバに保存されている最新の変更でローカルプロジェクト内のリソースを更新してプロジェクトを同期化します。プロジェクトのローカルバージョンをすでに持っていない場合は、[ローカルプロジェクト] ビューに作成されます。

### 注

ローカルプロジェクトで更新する前に、サーバ上の変更を確認する必要があります。詳細については、共有リソースの変更の結合に関する項目を参照してください。

- 3 プロジェクト同期ビューで変更したら、プロジェクトを同期化して変更をサーバに保存します。
- 4 リソースのロック解除

### 関連項目

- ・ 26 ページの[ローカルプロジェクトからの共有プロジェクトの作成](#)
- ・ 31 ページの[プロジェクトの同期](#)
- ・ 32 ページの[リソースのロック](#)
- ・ 32 ページの[リソースのロック解除](#)
- ・ 33 ページの[共有リソースへの変更の結合](#)

### 3.2.1 ローカルプロジェクトからの共有プロジェクトの作成

共有プロジェクトはローカルプロジェクトと同じ名前になります。この名前のプロジェクトが、リポジトリ内にあらかじめ存在していることはできません。

#### 注

既存の共有プロジェクトの名前を変更するには、[プロジェクトの同期]ビューで名前変更のコマンドを使用します。共有プロジェクトの名前変更の詳細については、関連項目を参照してください。


- 1 [ローカルプロジェクト]ビューで、共有するプロジェクトを選択します。右クリックし、[新しい共有プロジェクト]を選択します。
- 2 [セッションを開く] ダイアログボックスで、開くリポジトリシステムセッションを選択し、ユーザのシステム認証を入力します。  
[プロジェクトの同期]ビューが開き、ローカルプロジェクトと同じ名前の共有プロジェクトが表示されます。この時点では、共有プロジェクトは空です。
- 3 [プロジェクトの同期]ビューで、共有プロジェクトに保存するリソースを同期します。

#### 関連項目

- ・ 26 ページの[共有プロジェクトの名前の変更](#)
- ・ 175 ページの[セッションを開く](#)
- ・ 31 ページの[プロジェクトの同期](#)

### 3.2.2 共有プロジェクトの名前の変更

この手順に従って、リポジトリに存在する共有プロジェクトの名前を変更します。

- 1 共有プロジェクトが保存されるリポジトリシステム上のセッションで、プロジェクトの同期ビューを開きます。
- 2 [共有プロジェクト]リストボックスから共有プロジェクトを選択します。
- 3 [共有プロジェクト名の変更]アイコン  をクリックします。
- 4 リポジトリ内で一意の新しい名前を入力します。
- 5 [ローカルプロジェクト]ビュー内で、新しい名前のローカルプロジェクトを作成するには、プロジェクトを同期します。


オリジナルの名前を持つローカルプロジェクトは、新しい名前の共有プロジェクトに関連付けられていません。これらのローカルプロジェクト内のリソースは、新しく名前を付けた共有プロジェクトとは同期できなくなります。

#### 関連項目

- ・ 30 ページの[\[プロジェクト同期\]ビューを開く](#)

- ・ 31 ページの[プロジェクトの同期](#)」

### 3.2.3 共有プロジェクトの削除

- 1 共有プロジェクトが保存されるリポジトリシステム上のセッションで、プロジェクトの同期ビューを開きます。
- 2 [共有プロジェクト] リストボックスから共有プロジェクトを選択します。
- 3 [共有プロジェクトの削除] アイコン  をクリックします。

#### 注

共有プロジェクトを削除すると、元に戻すことはできません。

共有プロジェクトはリポジトリから削除されます。削除された共有プロジェクトに関連付けられているローカルプロジェクト内のリソースには影響しませんが、ローカルプロジェクトの同期ステータスは失われます。

## 3.3 プロジェクト同期について

プロジェクトを同期化するには、まず、[ローカルプロジェクト] ビューでプロジェクト内のリソースを、リポジトリサーバ上の関連する共有プロジェクトと比較します。同期化によって、追加されたリソース、削除されたリソース、リソース間の違いが検出されます。検出された違いに基づいて、ローカルおよび共有リソースを更新できます。

プロジェクトの同期ビューを使用してプロジェクトを同期化します。ビューでは、2 つの枠に同期情報が表示されます。



- ・ [共有プロジェクト] 枠には、サーバ上の共有プロジェクトのリソースが表示されます。ロックされているリソースの隣には鍵のアイコンが表示されます。リソースを最後に変更したユーザ、変更日、リソースをロックしたユーザとその日付など、サーバ上のリソースに関するその他の情報が提供されます。
- ・ [同期ステータス] 枠には各リソースのステータスが表示されます。ステータスは、ローカルおよび共有プロジェクトのリソースを比較することで決定されます。


異なる同期ステータスとその意味が表に表示されます。

ステータス	説明
ローカルで追加されました	リソースはローカルプロジェクトには追加されましたが、共有プロジェクトには追加されていません。
ローカルで変更されました	リソースは、ローカルプロジェクトで変更されましたが、最後の同期以来、共有プロジェクトでは変更されていません。
ローカルで削除されました	リソースはローカルプロジェクトで削除されましたが、共有プロジェクトにはまだ存在します。






ステータス	説明
サーバで追加されました	リソースはローカルプロジェクトにありませんが、共有プロジェクトに存在します。
サーバ上で変更されました	リソースは、共有プロジェクトで変更されましたが、最後の同期以来、ローカルプロジェクトでは変更されていません。
サーバ上で削除されました	リソースはローカルプロジェクトに存在しますが、共有プロジェクトで削除されました。
競合しています	以下のいずれかの状況で競合ステータスが発生します。 <ul style="list-style-type: none"><li>最後の同期以来、ローカルおよび共有プロジェクトのリソースに異なる変更が加えられた。</li><li>最後の同期以来、ローカルおよび共有プロジェクトに同じ名前のリソースが加えられた。</li><li>リソースがローカルプロジェクトでは変更されたが、共有プロジェクトから削除された。</li><li>リソースが共有プロジェクトでは変更されたが、ローカルプロジェクトから削除された。</li></ul>
同期化	リソースは同一です。

3つのコマンドを使ってリソースを同期化できます。同期するリソースを選択するときは、リソースまたはフォルダを個別に選択できます。以下の表は、考えられる同期化アクションをまとめたものです。

アイコン	コマンド	同期化アクション
	サーバから変更を取得	<p>選択されたリソース</p> <ul style="list-style-type: none"> <li>・ ステータスが[サーバで追加されました]の場合は、リソースがローカルプロジェクトに追加されています。</li> <li>・ ステータスが[サーバ上で変更されました]の場合は、リソースがローカルプロジェクトで更新されています。</li> <li>・ ステータスが[サーバ上で削除されました]の場合は、リソースがローカルプロジェクトで削除されています。</li> <li>・ ステータスが[競合しています]の場合は、サーバ上のリソース(変更、追加、削除されたかにかかわらず)は、ローカルプロジェクトで変更されたかにかかわらず、ローカルプロジェクトにコピーされます。</li> </ul> <p>他のすべてのステータスでは、アクションは取られません。</p> <p><b>注</b> ローカルプロジェクトで更新する前に、サーバ上の変更を確認する必要があります。詳細については、共有リソースの変更の結合に関する項目を参照してください。</p>
	変更をサーバに保存	<p>選択されたリソース</p> <ul style="list-style-type: none"> <li>・ ステータスが[ローカルで追加されました]の場合、リソースはサーバ上の共有プロジェクトに追加されています。</li> <li>・ ステータスが[ローカルで変更されました]の場合、リソースはサーバ上の共有プロジェクトで更新されています。</li> <li>・ ステータスが[ローカルで削除されました]の場合、リソースはサーバ上の共有プロジェクトで削除されています。</li> <li>・ ステータスが[競合しています]の場合は、ローカルプロジェクト上のリソース(変更、追加、削除されたかにかかわらず)は、共有プロジェクトで変更されたかにかかわらず、共有プロジェクトにコピーされます。</li> </ul> <p><b>注</b> リソースが他のユーザによってロックされている場合は、エラーメッセージが表示され、変更や削除はサーバでは実行されません。</p> <p>他のすべてのステータスでは、アクションは取られません。</p>

アイコン	コマンド	同期化アクション
	変更の復元	<p>選択したリソースについて、ローカルプロジェクトは、ステータスにかかわらず、サーバ上の共有プロジェクトで更新されます。</p> <p><b>注</b>  [変更の復元] では、リソースがローカルプロジェクトで作成されサーバ上でまだ保存されていない場合を除いて、[サーバから変更を取得]と同様にローカルプロジェクトが更新され、[変更の復元] では、新しいローカルリソースが削除され、[サーバから変更を取得] では新しいローカルリソースが維持されます。</p>



[同期ステータス] 枠のリソース一覧は、枠のツールバーのアイコンを使ってステータスを条件にフィルタできます。

	すべてのリソースを表示します。これにより、フィルタがクリアされ、ステータスにかかわらずすべてのリソースが表示されます。
	ステータスが [同期化] のリソースを表示/非表示にします。
	サーバに関して変更のあったローカルプロジェクトのリソースを表示/非表示にします。
	ステータスが [競合しています] のリソースを表示/非表示にします。
	ローカルプロジェクトに関して変更のあったサーバのリソースを表示/非表示にします。

#### 関連項目

- ・ 31 ページの[プロジェクトの同期](#)
- ・ 32 ページの[リソースのロック](#)
- ・ 32 ページの[リソースのロック解除](#)
- ・ 33 ページの[共有リソースへの変更の結合](#)

### 3.3.1 [プロジェクト同期] ビューを開く

- <sup>1</sup> [プロジェクト同期] ビューを開くには、ツールバーで [プロジェクト同期]  アイコンをクリックします。
- <sup>2</sup> [プロジェクト同期] ビューで [セッションの変更]  アイコンをクリックし、共有プロジェクトが保存されているリポジトリシステムでセッションを開きます。

ユーザの認証情報を入力した後、共有プロジェクトを管理し、同期するプロジェクトを [共有プロジェクト] リストボックスで選択します。



#### 関連項目

- ・ 175 ページの[セッションを開く](#)」
- ・ 27 ページの[プロジェクト同期について](#)」

### 3.3.2 プロジェクトの同期


プロジェクトを同期するには、プロジェクトが共有されている必要があります。

次のタスクを実行する場合、プロジェクトを同期します。

- ・ 共有プロジェクトに保存されている変更のあるローカルリソースを更新します。
  - ・ ローカルリソースに対して行った変更を共有プロジェクトに保存します。
  - ・ ローカルリソースを共有プロジェクトに保存されているコピーに戻します。
  - ・ 共有プロジェクトのローカルコピーを作成します。
- 1 共有プロジェクトが保存されるリポジトリシステム上のセッションで、プロジェクトの同期ビューを開きます。
  - 2 [共有プロジェクト] リストから共有プロジェクトを選択します。
  - 3 プロジェクト内のリソースの最新同期ステータスを見るには、[同期ステータス (ローカルプロジェクトと共有プロジェクトを比較)] というラベルのペインで、プロジェクトを展開し、最新表示アイコン  をクリックします。同期ステータスおよび可能なアクションの詳細については、プロジェクトの同期についての関連項目を参照してください。
  - 4 プロジェクトを同期します。
    - ・ 共有プロジェクト内で変更されたリソースが含まれるローカルプロジェクトを更新するには、一覧からリソースを選択して [サーバから変更を取得] アイコン  をクリックします。共有プロジェクトの名前を持つプロジェクトが [ローカルプロジェクト] ビューに存在しない場合、ローカルプロジェクトが作成されます。

#### 注


ローカルプロジェクトで更新する前に、サーバ上の変更を確認する必要があります。詳細については、共有リソースの変更の結合に関する項目を参照してください。

- ・ ローカルで行われた変更がある共有プロジェクトを更新するには、一覧からリソースを選択して [変更をサーバに保存] アイコン  をクリックします。

#### 注

別のユーザによってロックされているリソースは、サーバ上で更新できません。しかし、必要な場合は、どのユーザでもリソースのロックを解除できます。

ユーザがロックしているリソースでサーバを更新する場合、変更の同期操作ではリソースのロックは解除されません。サーバ上で明示的にこのリソースのロック解除を行う必要があります。

- ・ ローカルプロジェクト内のリソースをサーバに保存されているコピーに戻すには、リソースを選択して[変更を元に戻す] アイコン  をクリックします。

#### 関連項目

- ・ 27 ページの[プロジェクト同期について](#)
- ・ 30 ページの[\[プロジェクト同期\] ビューを開く](#)
- ・ 32 ページの[リソースのロック](#)
- ・ 32 ページの[リソースのロック解除](#)
- ・ 33 ページの[共有リソースへの変更の結合](#)

### 3.3.3 リソースのロック

リソースをロックするには、リソースが共有プロジェクト内にある必要があります。

他のデザイナーが[プロジェクト同期] ビューを開いている時に、そのリソースで作業中であることを知らせたい場合、リソースをロックします。

- 1 共有プロジェクトが保存されるリポジトリシステム上のセッションで、プロジェクトの同期ビューを開きます。
- 2 [共有プロジェクト] リストボックスから共有プロジェクトを選択します。
- 3 [共有プロジェクト] 枠で、プロジェクトを展開します。
- 4 リソースを右クリックし、[ロック] を選択します。

#### 注

ロックアクションは、リソースがローカルプロジェクトまたは共有プロジェクトのいずれにあっても、リソースコンテンツを更新しません。変更を保存するには、リソースを同期します。

#### 関連項目

- ・ 30 ページの[\[プロジェクト同期\] ビューを開く](#)
- ・ 27 ページの[プロジェクト同期について](#)

### 3.3.4 リソースのロック解除

サーバ上で変更を更新したため、他のデザイナーに完了したことを通知したい場合、リソースのロックを解除します。いったんリソースをロック解除すると、他のデザイナーが、リソースをロックしたり、変更のあるサーババージョンを更新したりすることができます。

#### 注

必要な場合、別のユーザがロックしたリソースをロック解除できます。



- 1 共有プロジェクトが保存されるリポジトリシステム上のセッションで、プロジェクトの同期ビューを開きます。
- 2 [共有プロジェクト] リストボックスから共有プロジェクトを選択します。
- 3 [共有プロジェクト] 枠で、プロジェクトを展開します。
- 4 リソースを右クリックし、[ロック解除] を選択します。

**注**

ロック解除アクションは、ローカルプロジェクトで変更があっても、サーバ上のリソースを更新しません。変更を保存するには、リソースを同期します。

**関連項目**

- ・ 30 ページの[\[プロジェクト同期\] ビューを開く](#)」
- ・ 27 ページの[プロジェクト同期について](#)」

### 3.3.5 共有リソースへの変更の結合

共有リソースを同期する際、サーバから変更を取得する前に、変更をレビューしてローカルリソースに適用する変更を決定したいことがあります。次の手順では、異なるリソース間の変更を手動で結合する方法について説明します。

たとえば、ローカルプロジェクト内にある "NewDatafoundation" というリソースで作業しているとします。このプロジェクトは、リポジトリ内で共有されています。プロジェクトを同級する場合、"NewDatafoundation" には [サーバ上で変更されました] または [競合しています] という同期ステータスがあります。

変更をレビューして手動で結合するには、次の操作を実行します。

- 1 ローカルプロジェクトで、"NewDatafoundation" を右クリックし、[コピー] を選択します。
- 2 ローカルプロジェクト内で再度右クリックし、[ペースト] を選択します。  
"NewDatafoundation" のコピーがローカルプロジェクト内に保存されます。
- 3 [プロジェクト同期] ビューで、"NewDatafoundation" を選択し、[サーバから変更を取得] を選択して同期します。
- 4 "NewDatafoundation" および "NewDatafoundation のコピー" の各リソース名をローカルプロジェクトでダブルクリックし、データファンデーションエディタで両方を開きます。  
各コピーがエディタの別のタブに開きます。
- 5 "NewDatafoundation" でのサーバからの変更と、"NewDatafoundation のコピー" でのローカルの変更とを比較します。
- 6 "NewDatafoundation" を開いているエディタのタブで、保持しないサーバ変更を削除し、保持するローカルでの変更を追加します。
- 7 エディタで、"NewDatafoundation" への変更を保存します。
- 8 [プロジェクト同期] ビューで、同期を最新表示します。"NewDatafoundation" を選択してから、[変更をサーバに保存] を選択して、サーバを更新します。

これで、結合された変更を確認すれば、ローカルプロジェクトから "NewDatafoundation のコピー" を削除することができます。

### 3.4 リソースの依存関係について

ローカルプロジェクトから削除したり、他のローカルプロジェクトに移動したり、更新するなどの変更をリソースに加えると、それに依存する他のリソースに影響を与えます。リソースを削除したり移動する前に影響の警告が表示されます。

変更の影響を理解し、作業を計画できるように、リソースとそのオブジェクトの間の依存関係を示すコマンドが存在します。

#### ローカルリソース間の依存関係

[ローカルの依存関係の表示] コマンドは、ローカルプロジェクトのリソース間の依存関係を表示します。

#### 例

〈データファンデーション A〉を変更したときに影響を受けるすべてのリソースを表示します。〈データファンデーション A〉の [ローカルの依存関係の表示] コマンドを選択します。

[依存リソース] タブには、〈ビジネスレイヤ X〉と、〈ビジネスレイヤ Y〉が表示されます。これらのビジネスレイヤには、〈データファンデーション A〉への参照が含まれ、〈データファンデーション A〉を削除したり変更すると無効な参照が含まれます。

[参照リソース] タブには〈接続 Z〉が表示されます。データファンデーションはこの接続を参照します。〈接続 Z〉を変更すると、〈データファンデーション A〉とその依存リソースに影響を与えることがあります。

#### オブジェクトとローカルリソースの間の依存関係

また、データファンデーションおよびビジネスレイヤエディタでは、[ローカルの依存関係の表示] コマンドを使った、特定のテーブル、列、またはビジネスレイヤオブジェクトに依存するローカルリソースを表示できます。

#### ローカルリソースとリポジトリリソース間の依存関係

[リポジトリの依存関係の表示] コマンドでは、選択したデータファンデーションまたはビジネスレイヤによって参照される特定のリポジトリに公開されたユニバースを表示します。

#### 関連項目

- ・ 34 ページの [ローカル依存関係の表示](#)
- ・ 35 ページの [リポジトリ依存関係の表示](#)
- ・ 94 ページの [ローカル依存関係の表示](#)

#### 3.4.1 ローカル依存関係の表示

選択したリソースに依存するローカルプロジェクトにリソースを表示するには、次の操作を実行します。

- 1 [ローカルプロジェクト] ビューで、依存関係を表示するリソースを選択します。
- 2 右クリックし、[ローカルの依存関係の表示.] を選択します。

[依存リソース] タブに、選択したリソースへの参照または依存を含む同じローカルプロジェクト内のリソースが、一覧表示されます。

[参照リソース] タブに、選択したリソースによって参照されている同じローカルプロジェクト内のリソースが、一覧表示されます。

#### 関連項目

- ・ 34 ページの[リソースの依存関係について](#)

### 3.4.2 リポジトリ依存関係の表示

選択したリソースに依存するリポジトリ内のユニバースを表示するには、次の操作を実行します。

- 1 [ローカルプロジェクト] ビューで、リポジトリ内で公開された依存リソースを表示するリソースを選択します。
- 2 右クリックし、[ローカルの依存関係の表示] を選択します。
- 3 リソースが公開されるリポジトリシステムのセッションを選択し、認証情報を入力します。

選択したリソースを参照するリポジトリ内の公開されたユニバースが一覧表示されます。

#### 関連項目

- ・ 175 ページの[セッションを開く](#)
- ・ 34 ページの[リソースの依存関係について](#)

### 3.5 公開されたユニバースの取得

公開されたユニバースを取得するには、ビジネスレイヤと参照リソースが保存される [ローカルプロジェクト] ビューにプロジェクトがある必要があります。

- 1 公開ユニバースウィザードの取得の開始

- ・ リポジトリから公開されたユニバースを取得するには、[ローカルプロジェクト] ビューでプロジェクトを選択して右クリックし、[ユニバースの取得] > [リポジトリから] を選択します。

[リポジトリリソース] ビュー内のリポジトリからユニバースを取得することもできます。ユニバースを右クリックし、[ユニバースの取得] を選択します。

**注**

デフォルトでは、リソースはローカルプロジェクト内に取得されます。取得されたデータファンデーションやビジネスレイヤを開くときに、ユーザは CMS 認証の入力を求められます。これにより、ローカルでのセキュリティが保護されます。

このローカルセキュリティ要件を削除するには、リポジトリでユニバースを選択している状態で [すべてのユーザ用に保存] チェックボックスをオンにします。

- ・ ローカルフォルダから公開されたユニバースを取得するには、[ローカルプロジェクト] ビューでプロジェクトを選択して右クリックし、[ユニバースの取得] > [ローカルフォルダから] を選択します。
- 2 ウィザードページの指示に従います。特定のページでの操作方法については、ヘルプアイコンをクリックしてください。

ウィザードが終了すると、ビジネスレイヤと依存リソース (接続、接続ショートカット、データファンデーション) がローカルプロジェクトに作成され、編集できるようになります。

**関連項目**

- ・ 175 ページの[セッションを開く](#)」
- ・ 171 ページの[リポジトリフォルダの選択](#)」
- ・ 24 ページの[ローカルプロジェクトの作成](#)」

## .unv ユニバースの変換

### 4.1 .unv ユニバースおよび .unx ユニバース

インフォメーションデザインツールの [.unv ユニバースの変換] コマンドを使用して、他の SAP BusinessObjects ユニバースデザインツールで作成されたユニバースと、以前のバージョンで作成されたユニバースを変換できます。次に、インフォメーションデザインツールで作成したユニバースと同様に、変換されたユニバースをローカルプロジェクトで 사용할 ことができます。

.unv ユニバースとは

.unv ユニバースは、たとえばユニバースデザイナーなどのすべての SAP Business Objects XI 3 のデザインツールで作成されたユニバースを参照しています。

以下の SAP Business Objects BI 4 デザインツールは .unv ユニバースを作成します。

- ・ ユニバースデザインツール (ユニバースデザイナーの新名称)
- ・ ユニバースデザインツールデスクトップエディション (ユニバースデザイナーパーソナルの新名称)

ユニバースは、<ユニバース名>.unv というファイル名でローカルフォルダまたはリポジトリに保存されます。

.unx ユニバースとは

インフォメーションデザインツールを使用してユニバースを公開すると、ユニバースは <ユニバース名>.unx というファイル名で保存されます。これは .unx ユニバースと呼ばれます。[.unv ユニバースの変換] コマンドは、.unv ユニバースを .unx ユニバース形式に変換します。

関連項目

- ・ 37 ページの [.unv ユニバースの変換について](#)

### 4.2 .unv ユニバースの変換について

インフォメーションデザインツールで .unv ユニバースを操作するには、これらを変換する必要があります。

変換可能な .unv ユニバースについて

以下の種類の .unv ユニバースを変換できます。

- ・ SAP BusinessObjects BI 4 ツール、ユニバースデザインツールまたはユニバースデザインツールデスクトップエディションを使用して作成されたリレーショナルユニバース。
- ・ SAP BusinessObjects Enterprise XI 3 デザインツールを使用して作成されたリレーショナルユニバース。

**注**

リポジトリに保存されているバージョン XI 3 に作成されたユニバースを変換できるようにするには、アップグレードマネジメントツールを使用してユニバースをアップグレードする必要があります。詳細については、『SAP Business Objects Business Intelligence platform アップグレードガイド』を参照してください。

以下の種類の .unv ユニバースは変換できません。

- ・ OLAP ユニバース
- ・ ストアドプロシージャユニバース
- ・ Data Federator データソースに基づくユニバース
- ・ Javabeen ユニバース

**注**

ビジネスビューマネージャ XI 3 で作成されたビジネスビューを、バージョン BI 4 のレポートングツールと互換性のある形式に変換することはできません。

**.unv ユニバースの変換方法**

.unv ファイルの変換方法は、ユニバースを作成するために使われたツールのソフトウェアバージョン、ユニバースがローカルに保存されているか、またはリポジトリに保存されているかによって異なります。次の表では、異なる変換シナリオで従うための手順について説明しています。変換手順の詳細については、関連項目を参照してください。

変換するユニバース	ワークフロー
XI 3 デザインツールを使用してリポジトリに保存された .unv ユニバース。	<p>最初に、アップグレードマネジメントツールを使用して、リポジトリ内のユニバースを最新バージョンにアップグレードします。</p> <p>インフォメーションデザインツールで、リポジトリ内の .unv ユニバースを変換するための手順に従います。</p> <p>変換によって、関連付けられたユニバースと接続権限を持つ同一の .unx ユニバースがリポジトリ内に作成されます。</p>
ユニバースデザインツールリリース BI 4.0 以降を使用してリポジトリに保存された .unv ユニバース。	<p>インフォメーションデザインツールで、リポジトリ内の .unv ユニバースを変換するための手順に従います。</p> <p>変換によって、関連付けられたユニバースと接続権限を持つ同一の .unx ユニバースがリポジトリ内に作成されます。</p>

変換するユニバース	ワークフロー
<p>バージョン XI 3 以降のデザインツールを使用して作成され、ローカルに保存されている .unv ユニバース。</p> <p><b>注</b> ローカルに保存されているユニバースは、すべてのユーザ用に保存されたセキュリティで保護されていないユニバースを参照します。</p>	<p>インフォメーションデザインツールで、ローカルに保存された .unv ユニバースを変換するための手順に従います。</p> <p>変換では、ローカルプロジェクトに同一のユニバースリソース（データファンデーション、ビジネスレイヤ、およびローカル接続）を作成します。</p> <p>.unx ユニバースを作成するためにビジネスレイヤを公開します。</p>

#### .unv ユニバースの変換後

.unv ユニバースを変換する場合、.unv ユニバースは維持されます。ユニバースに基づく SAP BusinessObjects クエリおよびレポートツール内のドキュメントは、そのまま .unv ユニバースにリンクされます。これで、変換されたユニバースに依存するドキュメントを変更する前に、チェックおよびテストする機会が得られます。

.unv ユニバースの一部の機能は、.unx ユニバースではさまざまに実装されます。いったんユニバースを変換すると、インフォメーションデザインツールでローカルプロジェクト内のユニバースリソースを編集し、不一致をチェックおよび修正して、新しいユニバースの機能を利用できるようになります。サポートされている機能の説明と、それらの .unx ユニバースでの実装方法の詳細については、関連項目を参照してください。

ユニバースの変換後に、データファンデーションの構造を最新表示にして、ユニバースで整合性のチェックを実行することをお勧めします。変換されたユニバースにおける整合性のチェックエラーの解決方法については、関連項目を参照してください。

#### 関連項目

- 45 ページの [リポジトリでの .unv ユニバースの変換](#)
- 46 ページの [ローカルに保存された .unv ユニバースの変換](#)
- 39 ページの [.unv ユニバースの変換時にサポートされている機能](#)
- 44 ページの [.unv ユニバース変換後の整合性チェックのエラーの解決方法](#)
- 37 ページの [.unv ユニバースおよび .unx ユニバース](#)

### 4.3 .unv ユニバースの変換時にサポートされている機能

インフォメーションデザインツールを使用して .unv ユニバースを変換すると、変換処理によって、変換されたユニバース内に同じ機能が作成されます。以下の表は、サポートされている .unv ユニバース機能と、.unx ユニバース内での実装方法を示しています。特定の機能については、最善の変換結果を得るためのヒントを説明しています。

元の .unv ユニバースの機能	変換後の .unx ユニバースの機能
ユニバーススキーマ	<p>ユニバーススキーマ内の以下のオブジェクトが、データファンデーション内に作成されます。</p> <ul style="list-style-type: none"> <li>・ テーブル</li> <li>・ エイリアステーブル</li> <li>・ 派生テーブル (ネストされた派生テーブルを含む)</li> <li>・ 結合 (ショートカット結合を含む)</li> <li>・ 自己結合 (列フィルタに変換)</li> </ul> <p><b>ヒント</b> 自己結合式内の @Prompt は、変換後に手動介入が必要な場合があります。整合性のチェックのエラー修正に関連する項目を参照してください。</p> <ul style="list-style-type: none"> <li>・ コンテキスト</li> </ul> <p><b>ヒント</b> コンテキストは、明示的に含まれているか、除外されているすべての結合と一緒に変換されます。データファンデーションエディタでは、シンプルなコンテキスト機能を利用できます。中性結合を使用するスキーマのあいまいな部分に対して手動でコンテキスト定義を制限できます。コンテキストの詳細については、関連項目を参照してください。</p>
ユニバースアウトライン	<p>ユニバースアウトライン内の以下のオブジェクトが、すべてのプロパティを含めてビジネスレイヤ内に作成されます。</p> <ul style="list-style-type: none"> <li>・ クラスおよびサブクラス (フォルダに変換)</li> <li>・ ディメンション時間階層の場合、ディメンションは階層内のアクティブな各レベルに対して作成されます。</li> <li>・ メジャー (集計関数を含む)</li> <li>・ 詳細 (属性に変換済み)</li> <li>・ 条件 (必須フィルタのプロパティを含むフィルタに変換)</li> </ul>
複数言語ユニバース	<p>すべての翻訳された文字列、言語設定、およびロケールの設定は変換されます。</p>



元の .unv ユニバースの機能	変換後の .unx ユニバースの機能
リンクされたユニバース (コアおよび派生)	<p>コアユニバースはどのような .unv ユニバースでも変換されます。コアユニバースの変換では、依存する派生ユニバースの変換は発生しません。</p> <p>派生ユニバースは、コアユニバースにリンクされています。派生ユニバースを変換する場合、変換には、派生ユニバースがリンクされているすべてのコアユニバースが自動的に含まれます。変換前に、別の手順としてコアユニバースを変換しておく必要はありません。</p> <p>変換されたユニバースのデータファンデーションには、すべてのコアユニバースからすべてのテーブルと結合、および派生ユニバースで定義されていたあらゆるテーブル、結合、またはコンテキストが含まれます。</p> <p>ビジネスレイヤには、派生ユニバースで定義されていたクラス、オブジェクト、および条件を含むすべてのコアユニバースからのすべてのクラス、オブジェクト、および条件が含まれます。</p>
ユニバースコントロールパラメータ: クエリ制限	クエリ制限は変換され、ビジネスレイヤで編集できます。
ユニバース SQL パラメータ: SQL 制限	<p>クエリ、複数の SQL 文、およびデカルト積コントロールは変換されます。</p> <p>[デカルト積を許可する] および [各コンテキストに対して複数の SQL 文を生成] の制限は、データファンデーションで編集できます。その他のすべての SQL 制限はビジネスレイヤで編集します。</p>
ストラテジー	カスタムストラテジーは .unx ユニバースではサポートされていません。
ユニバースパラメータ: SQL 生成パラメータ	<p>PRM ファイルまたはユニバースパラメータ内の SQL 生成パラメータ設定に対するカスタマイズは、変換されません。インフォメーションデザインツールを使用して、カスタマイズされた値を変換済み PRM ファイルに追加し、変換済みユニバース内のユニバースパラメータ設定をカスタマイズできます。</p> <p><b>ヒント</b> データファンデーションプロパティおよびビジネスレイヤプロパティで SQL 生成パラメータに対するカスタム設定を確認およびリセットします。SQL パラメータの設定の詳細については、関連項目を参照してください。</p>

元の .unv ユニバースの機能	変換後の .unx ユニバースの機能
@ 関数	<p>次の @関数に変換されます。</p> <ul style="list-style-type: none"> <li>・ @Aggregate_Aware</li> <li>・ @Prompt</li> <li>・ @DerivedTable</li> <li>・ @Select</li> <li>・ @Variable</li> <li>・ @Where</li> </ul> <p>すべての関数の構文がサポートされています。</p> <p><b>ヒント</b> @Prompt 関数には、指定されたパラメータを使用するための新しい別の構文があります。詳細については、関連項目を参照してください。</p>
プロンプト	<p>ビジネスレイヤのディメンションおよびメジャー式、およびデータファンデーションの SQL 式内の @Prompts はサポートされています。</p> <p>変換時には、ビジネスレイヤでプロンプトに対して指定されたパラメータを、自動的に作成するためのオプションがあります。</p> <p>データファンデーション内の @Prompt 式は変換されません。自己結合式内の @Prompt は、変換後に手動介入が必要な場合があります。整合性のチェックのエラー修正に関連する項目を参照してください。</p> <p><b>ヒント</b> インフォメーションデザインツール内のパラメータおよび値の一覧は、これらが参照するオブジェクトとは無関係に定義できます。そのため、1 つ以上のビジネスレイヤオブジェクト内の指定されたパラメータまたは値の一覧を参照できます。</p>
値の一覧	<p>指定された値の一覧が、値の一覧を指定するディメンションおよびメジャーオブジェクトのビジネスレイヤに作成されます。</p>
アクセス制限	<p>リポジトリでユニバースを変換する場合、ユニバースのアクセス制限は、セキュリティエディタを使用して編集可能なセキュリティプロファイルに変換されます。</p> <ul style="list-style-type: none"> <li>・ アクセス制限 (オブジェクト制限を除く) は、データセキュリティプロファイル内の設定に変換されます。</li> <li>・ オブジェクトのアクセス制限は、ビジネスセキュリティプロファイル内のクエリの作成および表示データ設定に変換されます。</li> </ul> <p><b>ヒント</b> ビジネスセキュリティプロファイルでは、メタデータをデータとは別に保護することができます。たとえば、あるユーザが該当データの表示を許可されていなくても、そのユーザに対してクエリの作成を許可することなどが可能です。インフォメーションデザインツールのユニバースセキュリティの詳細については、関連項目を参照してください。</p>

元の .unv ユニバースの機能	変換後の .unx ユニバースの機能
セキュリティの割り当ておよび優先順位	<p>リポジトリでユニバースを変換する場合、ユーザおよびグループ割り当ては変換されます。</p> <p><b>ヒント</b> セキュリティエディタで、1 つ以上のセキュリティプロファイルをユーザまたはグループに割り当てられる機能を使用できます。</p> <p>アクセス制限のグループの優先順位は変換されます。</p> <p><b>注</b> .unv ユニバースの場合、ユーザが異なる複数のグループに属している場合、ユーザがアクセス制限を割り当てられていないと、グループに対して割り当てられた優先順位が、ユーザの継承するアクセス制限を決定します。変換されたユニバースでは、優先順位はグループではなくデータセキュリティプロファイルに割り当てられます。グループに割り当てられたプロファイルの優先順位が、ユーザに割り当てられたプロファイルの優先順位よりも高い場合、グループのプロファイルが使用されます。</p>
接続	<p>リポジトリでユニバースを変換する場合、同じセキュリティリレーショナル接続が .unv と .unx ユニバースの両方によって使用されます。変換されたユニバースをローカルプロジェクトに取得する場合、リポジトリ内のセキュリティ接続を参照する接続ショートカットが作成されます。</p> <p><b>ヒント</b> リレーショナル接続は、ユニバースデザインツールでもインフォメーションデザインツールでも作成して共有できます。接続は、リポジトリ内の同じ [Connections] フォルダで公開されます。</p> <p>ローカルに保存されている (セキュリティで保護されていない) ユニバースを変換する場合、個人用および共有接続は、ローカル接続に変換されます。</p>

#### 関連項目

- ・ 44 ページの [.unv ユニバース変換後の整合性チェックのエラーの解決方法](#)」
- ・ 70 ページの [データファンデーションエディタについて](#)」
- ・ 90 ページの [コンテキストについて](#)」
- ・ 209 ページの [SQL 生成パラメータについて](#)」
- ・ 205 ページの [関数のリファレンス](#)」
- ・ 103 ページの [ビジネスレイアエディタについて](#)」
- ・ 177 ページの [ユニバースセキュリティの概要](#)」

## 4.4 .unv ユニバース変換後の整合性チェックのエラーの解決方法

.unv ユニバースの変換後に、変換されたユニバースのインフォメーションデザインツールで整合性のチェックを実行することをお勧めします。整合性のチェック結果内の特定のエラーは、以下で説明する最善の実践策に従って解決できます。

### 列のデータ型のエラー

変換直後にデータファンデーションの構造を最新表示します。これにより、整合性チェックでのデータ型エラーを回避します。

### @Prompt のある自己結合のエラー

.unv ユニバース内の結合式に、オブジェクトを参照している値の一覧のある @Prompt が含まれる場合、変換された結合はデータファンデーションで再加工する必要があります。2つの可能性のある解決方法の手順について説明します。説明には、以下の例を使用します。

.unv ユニバースには、言語というプロンプトのある dimProductStrings テーブルに自己結合が含まれます。自己結合式は、dimProductStrings.LanguageID= @Prompt('言語','N','言語¥言語 Id',mono,constrained) です。

ユニバースの変換後に、データファンデーションには dimProductStrings テーブルに列フィルタが含まれます。フィルタの結合式には @Prompt が含まれます。

最初の解決方法は、入力を求められるパラメータと値の一覧 (LOV) をデータファンデーションで作成することです。

- 1 データファンデーションを開くには、[ローカルプロジェクト]ビューにあるデータファンデーション名をダブルクリックします。
- 2 [パラメータおよび値の一覧] タブをクリックします。
- 3 データファンデーションの [値の一覧] ペインで、言語用のカスタム SQL に基づく LOV を定義します。たとえば、SELECT "LANGUAGES"."LANGUAGEID", "LANGUAGES"."LANGUAGECODE" FROM "LANGUAGES"。
- 4 データファンデーションの [パラメータ] ペインで、言語用のパラメータを定義します。[ユーザへのプロンプト] オプションを選択し、それを言語の LOV に関連付けます。
- 5 データファンデーションで、dimProductStrings テーブル内の列フィルタを編集します。結合式が新しく入力を求められるパラメータを参照するように変更します。たとえば、dimProductStrings.LanguageID= @Prompt(Language) となります。
- 6 データファンデーションを保存して閉じます。

2 番目の解決方法は、ビジネスレイヤ内の必須フィルタを使用します。

- 1 データファンデーションをエディタで開き、@Prompt を含む dimProductStrings テーブルの列フィルタを削除します。
- 2 データファンデーションを保存して閉じます。
- 3 エディタでビジネスレイヤを開きます。
- 4 ビジネスレイヤの [値の一覧] ペインで、言語用のカスタム SQL に基づき LOV を定義します。たとえば、SELECT "LANGUAGES"."LANGUAGEID", "LANGUAGES"."LANGUAGECODE" FROM "LANGUAGES"。
- 5 ビジネスレイヤの [パラメータ] ペインで、言語用のパラメータを定義します。デフォルトオプションの [ユーザへのプロンプト] を保持し、それを言語の LOV に関連付けます。

- 6 ビジネスレイヤにおいて、製品に関連付けられたフォルダで、言語の入力を求められるパラメータを参照する式があるフィルタを作成します。たとえば、dimProductStrings.LanguageID= @Prompt(Language) となります。
- 7 フィルタ定義の[プロパティ]タブで、[クエリで必須としてフィルタを使用する]ボックスをオンにします。[フォルダに適用]の[フィルタ連結範囲]を選択します。
- 8 ビジネスレイヤを保存して終了します。


#### 関連項目

- ・ 94 ページの[データファンデーションの最新表示について](#)
- ・ 70 ページの[データファンデーションエディタについて](#)
- ・ 93 ページの[データファンデーション内のパラメータと値の一覧について](#)
- ・ 83 ページの[列フィルタについて](#)
- ・ 103 ページの[ビジネスレイヤエディタについて](#)
- ・ 124 ページの[パラメータについて](#)
- ・ 126 ページの[値の一覧について](#)
- ・ 112 ページの[フィルタについて](#)

## 4.5 リポジトリでの .unv ユニバースの変換

変換される .unv ユニバースは、インフォメーションデザインツールと互換性のあるリポジトリに保存する必要があります。 .unv ユニバースが SAP BusinessObjects BI 4.0 より前のバージョンのデザインツールで作成された場合、まず、アップグレードマネジメントツールを使用してユニバースをアップグレードする必要があります。ユニバースのアップグレードの詳細については、『SAP BusinessObjects アップグレードガイド』を参照してください。

変換された .unx ユニバースをローカルプロジェクトに取り込んで作業をする場合、まず、[ローカルプロジェクト]ビューにローカルプロジェクトフォルダが存在する必要があります。

- 1 インフォメーションデザインツールで、[ファイル] > [.unv ユニバースの変換] を選択します。
- 2 [.unv ユニバースの変換] ダイアログボックスで、[リポジトリから .unv ユニバースを選択] アイコン  をクリックします。
- 3 .unv ユニバースが保存されるリポジトリでセッションを開き、ユニバースを選択して [OK] をクリックします。
- 4 [出力先リポジトリフォルダ] フィールドの横にある [参照] ボタンをクリックし、変換された .unx ユニバースを保存するリポジトリ内のフォルダを選択します。
- 5 変換された .unx ユニバースをローカルプロジェクトに取り込んで作業をする場合、[出力先ローカルプロジェクトフォルダ] フィールドの横にある [参照] ボタンをクリックして、プロジェクトフォルダを選択し、[OK] をクリックします。
- 6 プロンプトの指定されたパラメータを作成するために変換を行う場合、[@prompt 式をパラメータ指定済みユニバースに自動変換] ボックスのチェックをオンにします。指定済みパラメータの詳細については、関連項目を参照してください。

- 7 変換されたユニバースをローカルプロジェクトに取り込み、すべてのユーザがリポジトリ認証を入力しなくてもユニバースリソースを開けるように、ローカルセキュリティ要件を削除する場合、[すべてのユーザ用に保存] ボックスをオンにします。
- 8 [OK] をクリックすると変換が始まります。


変換後に、データファンデーションの構造を最新表示して、変換での問題を検出するためにユニバースで整合性のチェックを実行することをお勧めします。整合性のチェックエラーの解決方法については、関連項目を参照してください。

#### 関連項目

- ・ 39 ページの[.unv ユニバースの変換時にサポートされている機能](#)
- ・ 175 ページの[セッションを開く](#)
- ・ 24 ページの[ローカルプロジェクトの作成](#)
- ・ 124 ページの[パラメータについて](#)
- ・ 94 ページの[データファンデーションの最新表示について](#)
- ・ 155 ページの[\[整合性のチェック\] について](#)
- ・ 44 ページの[.unv ユニバース変換後の整合性チェックのエラーの解決方法](#)

## 4.6 ローカルに保存された .unv ユニバースの変換

[ローカルプロジェクト] ビューで、変換されたユニバースのリソースの保存先となるローカルプロジェクトフォルダを定義する必要があります。

- 1 インフォメーションデザインツールで、[ファイル] > [.unv ユニバースの変換] を選択します。
- 2 [.unv ユニバースの変換] ダイアログボックスで、[ローカルファイルシステムから .unv ユニバースを選択] アイコン  をクリックして、変換するユニバースを選択します。
- 3 [出力先ローカルプロジェクトフォルダ] フィールドの横にある [参照] ボタンをクリックして、プロジェクトフォルダを選択し、[OK] をクリックします。
- 4 プロンプトの指定されたパラメータを作成するために変換を行う場合、[@prompt 式をパラメータ指定済みユニバースに自動変換] ボックスのチェックをオンにします。指定済みパラメータの詳細については、関連項目を参照してください。
- 5 [OK] をクリックすると変換が始まります。

変換では、指定されたローカルプロジェクトフォルダに同一のユニバースリソース (データファンデーション、ビジネスレイヤ、およびローカル接続) を作成します。

変換後に、データファンデーションの構造を最新表示することをお勧めします。

.unx ユニバースファイルを作成するためにビジネスレイヤを公開します。公開ウィザードでは、ユニバースで整合性のチェックを実行できます (推奨)。整合性のチェックエラーの解決方法については、関連項目を参照してください。

#### 関連項目

- ・ 39 ページの[.unv ユニバースの変換時にサポートされている機能](#)

- ・ 24 ページの[ローカルプロジェクトの作成](#)」
- ・ 124 ページの[パラメータについて](#)」
- ・ 94 ページの[データファンデーションの最新表示について](#)」
- ・ 170 ページの[ユニバースの公開](#)」
- ・ 155 ページの[\[整合性のチェック\] について](#)」
- ・ 44 ページの[.unv ユニバース変換後の整合性チェックのエラーの解決方法](#)」





## 接続の使用

### 5.1 接続について

接続は、名前の付いたパラメータのセットのことで、1 つまたは複数の SAP BusinessObjects アプリケーションがリレーショナルまたは OLAP データベースにアクセスする方法を定義します。接続はローカルファイルにできるほか、インフォメーションデザインツールでローカルショートカットから参照されるリポジトリ内のリモートオブジェクトにすることもできます。

ユーザは、次の目的で接続を使用します。

接続の使用目的	説明
データファンデーションのリレーショナルデータソース	1 つまたは複数のリレーショナル接続をデータファンデーションに関連付けて、データファンデーション上でビジネスレイヤを作成します。  ビジネスレイヤをユニバースとして公開する場合、接続およびデータファンデーションはユニバース内に統合され、ユニバースに対して実行されるクエリのデータを提供します。
ビジネスレイヤの OLAP データソース	OLAP データソースの場合、ビジネスレイヤを直接接続に関連付けます。ビジネスレイヤは、ユニバースとして公開されますが、接続はキューブへのダイレクトアクセスを提供します。
SAP NetWeaver BW BEx クエリへのアクセス	BEx クエリに接続できるように、[SAP BICS Client] ミドルウェアドライバを使用する SAP NetWeaver BW 接続を定義します。SAP BusinessObjects qクエリおよびレポーティングアプリケーションは、直接 BEx クエリに接続します。これらの接続を、ビジネスレイヤまたはユニバースのソースとして使用することはできません。詳細については、SAP NetWeaver BW の接続に関する関連項目を参照してください。

接続は、ローカルまたはセキュリティのいずれかです。インフォメーションデザインツールのローカルプロジェクトでローカル接続を作成します。ローカル接続は、ローカルファイルシステム上の独立したオブジェクトとなり、.cnx ファイルとして保存されます。

ユニバースの定義の一部として、または別のリソースとしてのいずれかで、接続をリポジトリに公開する場合、セキュリティ接続として [Connections] フォルダまたはサブフォルダに保存されます。接続ショートカットは、接続プロパティにアクセスできるローカルプロジェクトで作成され、また、別のデータファンデーションまたはビジネスレイヤに関連付けられるように接続を有効にします。

また、リポジトリリソースビューから、[リレーショナル接続の挿入] および [OLAP 接続の挿入] コマンドを使用して、リポジトリ内に直接セキュリティ接続を作成することもできます。

#### 関連項目

- ・ 50 ページの [ローカル接続について](#)
- ・ 51 ページの [セキュリティ接続について](#)
- ・ 52 ページの [SAP NetWeaver BW 接続について](#)
- ・ 53 ページの [リレーショナル接続の作成](#)
- ・ 62 ページの [OLAP 接続の作成](#)
- ・ 67 ページの [接続および接続ショートカットの編集について](#)

### 5.1.1 ローカル接続について

ローカルプロジェクトで作成されたすべての接続は、リポジトリに公開される前はローカルコネクションです。接続が公開されると、ローカル接続と同じパラメータを含む接続オブジェクトがリポジトリの [接続] フォルダかサブフォルダに作成されます。ローカルプロジェクトの接続のためのローカルショートカットが作成されます。ローカル接続と同じ方法でショートカットを使用できますが、ターゲット接続は保護されており、リポジトリ内のセキュリティ制限に従います。

ローカル接続は次の目的で使用されます。

- ・ データファンデーションとリレーショナルビジネスレイヤの作成時にリレーショナルデータソースにアクセスするため。

#### 注

複数のソースが有効化されたデータファンデーションを作成するには、セキュリティ接続を参照する必要があります。

- ・ OLAP ビジネスレイヤの作成時に OLAP キューブにアクセスするため。
- ・ ターゲットデータベースに対してクエリを実行して、ビジネスレイヤでの変更をテストするか、値の一覧を作成するため。

ローカル接続は、制限されたセキュリティ、またはセキュリティがないため、インフォメーションデザインツールを実行中のマシンにアクセス権のあるすべてのユーザが使用できます。

ローカル接続は主に、データファンデーションおよびビジネスレイヤ開発のオーサリングフェーズで使用されます。ユニバースとしてビジネスレイヤがリポジトリに公開されると、接続がセキュリティ保護されます。接続ショートカットはローカルで使用可能ですが、公開された接続の編集は、適切なリポジトリシステム認証を使用するのみ実行できます。

### 5.1.2 セキュリティ接続について

セキュリティ接続とはリポジトリ内で作成されたか、公開された接続のことです。これはリポジトリ内の専用の [Connections] フォルダに保存されます [Connections] フォルダにサブフォルダを作成して、リポジトリ内に保存された接続を整理することができます。

セキュリティ接続は、ローカルファイルシステムにはコピーできませんが、[ローカルプロジェクト] ビューで接続ショートカットとして使用できます。接続ショートカットは、リポジトリのセキュリティ接続を参照するオブジェクトです。ショートカットは、ローカル接続と同じように使用できますが、接続プロパティはリポジトリに接続することによってのみ変更できます。

セキュリティ接続および接続ショートカットは次の目的で使用されます。

- ・ リポジトリに公開されたユニバースのデータを取得するため。
- ・ データベースミドルウェアに直接アクセスする SAP BusinessObjects レポート製品のデータを取得するため。
- ・ データファンデーションまたは OLAP ビジネスレイヤを作成するときのデータソースとするため。

ローカル接続をリポジトリに公開するか、リポジトリで直接接続を作成することにより、セキュリティ接続を作成できます。

セキュリティ接続は、リポジトリ内で次の標準セキュリティ制限に従います。

- ・ ユーザは認証される必要があります。
- ・ ユーザ権限は、接続または接続プロパティへのアクセスを許可するか、拒否するかをユーザレベルで定義できます。
- ・ 認証されたユーザのみが、接続を共有または使用することができます。

関連項目

- ・ 51 ページの [接続ショートカットについて](#)

### 5.1.3 接続ショートカットについて

接続ショートカットは、リポジトリのセキュリティ接続を参照するオブジェクトです。ショートカットは .cns ファイルとしてローカルファイルシステムに保存されます。ショートカットには、リポジトリアドレス、ポート番号、接続の種類 (OLAP またはリレーショナル)、およびサーバ上の接続を識別するための ID が含まれます。

接続ショートカットは、リポジトリ内に保存されている接続を使用するデータファンデーションまたはビジネスレイヤを作成、または変更するときに使用します。

接続ショートカットは次の 2 つの方法で作成できます。

- ・ リポジトリにローカル接続を公開する。

- ・ リポジトリリソースビューで既存のセキュリティ接続からショートカットを作成する。

#### 関連項目

- ・ 171 ページの[ローカル接続のリポジトリへの公開](#)
- ・ 67 ページの[接続ショートカットの作成](#)

### 5.1.4 SAP NetWeaver BW 接続について

インフォメーションデザインツールで、以下の目的で SAP NetWeaver BW 接続を定義できます。

目的	手順
SAP NetWeaver BW BEx クエリを SAP BusinessObjects クエリおよびレポーティングアプリケーションで使用可能にする。	<ol style="list-style-type: none"> <li>1 新しい OLAP 接続ウィザードを使用して、OLAP 接続を定義します。SAP NetWeaver BW [SAP BICS Client] ミドルウェアドライバを選択します。</li> <li>2 クエリおよびレポーティングアプリケーションからアクセス可能なリポジトリに、接続を公開します。</li> </ol>
複数ソースユニバースに SAP NetWeaver BW への接続を含める。	<ol style="list-style-type: none"> <li>1 [リポジトリリソース] ビューで 新しいリレーショナル接続ウィザードを使用して、リポジトリでリレーショナル接続を直接定義します。SAP NetWeaver BW ミドルウェアドライバを選択します。</li> <li>2 [新しいデータファンデーション] ウィザードを使用して、複数のソースが有効化されたデータファンデーションを作成します。複数のソースが有効化されたデータファンデーションの種類を必ず選択します。</li> <li>3 ビジネスレイヤの新規作成ウィザードを使用して、データファンデーションにビジネスレイヤを作成します。</li> <li>4 クエリおよびレポーティングアプリケーションからアクセス可能なリポジトリに、ビジネスレイヤを公開します。</li> </ol> <p><b>注</b> クエリおよびレポーティングアプリケーションのユーザが、SAP NetWeaver BW で複数のソースが有効化されたユニバースにアクセスするのに必要な認証についての詳細は、SAP ノート 1465871 を参照してください。</p>

#### 関連項目

- ・ 62 ページの[OLAP 接続の作成](#)
- ・ 99 ページの[ビジネスレイヤの作成](#)
- ・ 53 ページの[リレーショナル接続の作成](#)
- ・ 73 ページの[複数のソースが有効化されたデータファンデーションについて](#)

- ・ 169 ページの[リソースの公開について](#)

## 5.2 リレーショナル接続の作成

新しいリレーショナル接続ウィザードを使用して、リレーショナルデータソースにローカル接続およびセキュリティ接続を作成します。

ローカル接続を作成する前に、[ローカルプロジェクト] ビューにプロジェクトがある必要があります。ローカルプロジェクトの作成の詳細については、関連項目を参照してください。

### 注

SAP NetWeaver BW ソースおよび SAS ソースへのリレーショナル接続を、直接リポジトリ内にセキュリティ接続として作成する必要があります。

- 1 次のいずれかを実行します。
  - ・ ローカル接続を作成するには、[ローカルプロジェクト] ビューからプロジェクトフォルダを選択します。  
[ファイル] > [新規] > [リレーショナル接続] を選択します。
  - ・ セキュリティ接続を作成するには、[リポジトリリソース] ビューで、セキュリティ接続を作成するリポジトリでセッションを開きます。リポジトリ内の接続フォルダまたはサブフォルダを右クリックし、[リレーショナル接続の挿入] を選択します。
- 2 新規リレーショナル接続ウィザードの手順に従って、以下の情報を入力します。
  - ・ 接続名
  - ・ ターゲットデータベースのミドルウェア
  - ・ リレーショナルデータベースに接続するための認証パラメータ
  - ・ 接続を最適化する接続パラメータ

個々の手順で支援が必要な場合は、ウィザードダイアログボックスでヘルプアイコンをクリックしてください。

### 関連項目

- ・ 24 ページの[ローカルプロジェクトの作成](#)
- ・ 173 ページの[リポジトリのリソースの管理について](#)
- ・ 53 ページの[接続の名前](#)

### 5.2.1 接続の名前

この節では、新しいリレーショナル接続ウィザードのリソース名のページについて説明します。

接続の名前を付けて、データソースの説明を入力することができます。名前と説明は、接続のプロパティとして使用でき、いつでも編集することができます。

プロパティ	説明
リソース名	接続名。このフィールドは、必須です。
説明	データソースについて説明する情報です。この情報は、接続を複数のデータファンデーションで使用する場合に役立ちます。これはオプションの情報です。

名前情報を入力したら、[次へ] をクリックしてウィザードを続行します。

#### 関連項目

- ・ 62 ページの[ミドルウェアドライバの選択](#)」

## 5.2.2 データソースパラメータの設定

接続パラメータは、接続を定義しているデータソースの種類によって異なります。接続パラメータの詳細については、関連トピックのからリンクを選択してください。

#### 関連項目

- ・ 54 ページの[\[ログインパラメータ\]ダイアログボックス](#)」
- ・ 56 ページの[\[設定パラメータ\]ダイアログボックス](#)」
- ・ 58 ページの[SAP HANA 接続のパラメータ](#)」
- ・ 58 ページの[SAP NetWeaver BW データソースの接続パラメータ](#)」
- ・ 60 ページの[SAS データソースの接続パラメータ](#)」

### 5.2.2.1 [ログインパラメータ]ダイアログボックス

新規接続ウィザードの[ログインパラメータ]ダイアログボックスには、次のパラメータが含まれています。

#### 注

ユニバース接続マネージャ (UCM) から接続ウィザードにアクセスしているか、CMS を使用せずにオフラインモードでユニバースデザインツールで作業している場合、使用できる認証モードは、[指定したユーザ名とパスワードを使用する] のみです。

パラメータ	説明
認証モード	<ul style="list-style-type: none"> <li>指定したユーザ名とパスワードを使用する: ログインの詳細を認証として使用します。</li> <li>BusinessObjects 認証マップを使用する: レポートを最新表示するには、SAP BusinessObjects アカウントに関連付けられたデータベースユーザパスワードが要求されます。これは dbuser パラメータおよび dbpass パラメータを使用して設定します。これらは管理レベルで設定されます。このオプションの設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> <li>表示時のレポートの最新表示にシングルサインオンを使用する: このオプションを選択すると、CMS へのアクセスに使用したユーザ名とパスワードが、自動的にデータベースへのログインパラメータとして使用されます。シングルサインオン (SSO) の設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> </ul>
ユーザ名	[認証モード] が [指定したユーザ名とパスワードを使用する] の場合、データベースにアクセスするユーザ名。
パスワード	[認証モード] が [指定したユーザ名とパスワードを使用する] の場合、データベースにアクセスするパスワード。
データソース (<ホスト>:<ポート>)	データソースの詳細。
データベース	データベース名。
エイリアス	DB2 接続の場合は、DB2 データソースマネージャの中で作成したデータベースに関するエイリアスです。
サーバ	<p>ODBC 接続の場合は、データベースをホストしているマシンの名前です。</p> <p>JDBC 接続の場合は、データベースに接続するために使用する JDBC URL です。</p> <p>Oracle 接続の場合は、サーバの一覧も指定できます。</p>
カタログ	接続するカタログの名前。

パラメータ	説明
URL	たとえば、javabeans コネクタの場合は、このパラメータは使用中の javabeans の URL を指定します。  <b>注</b> javabeans を接続準備ができた状態にしておき、『データアクセスガイド』の説明に従って javabeans を設定する必要があります。
Informix サーバ	Informix 接続の場合は、定義した Informix サーバの名前です。
ネットサービス	Oracle 接続の場合は、Oracle Net service。
データソース名	ODBC 接続の場合は、オペレーティングシステムのデータソースマネージャで定義したデータソースの名前です。
サービス	Oracle 接続の場合は、Oracle Net Manager の中で定義したサービスです。

### 5.2.2.2 [設定パラメータ]ダイアログボックス

[設定パラメータ]ダイアログボックスには、デフォルトの設定オプションを上書きするために設定できるパラメータが含まれています。これらの設定パラメータは次の設定を上書きします。

- ・ cs.cfg ファイルに設定されている、対応するパラメータ
- ・ <ドライバ>.sbo ファイルに設定されている、対応するパラメータ

次のパラメータを設定できます。

接続プールモード	ドロップダウンリストから、接続の維持に使用する方法を選択します。接続プールを使用する場合にのみ必要です。
プールタイムアウト	上のフィールドで[プール維持時間]を選択した場合は、接続を維持する期間を指定します。



配列フェッチサイズ	<p>フェッチごとに取得できる最大行数を設定します。</p> <p>20 を指定すると、クエリ結果が 100 行の場合は 5 回のフェッチ動作を行います。</p> <p>「1」を入力すると配列フェッチは無効になり、行ごとにデータが取得されます。</p> <p><b>注</b> 配列フェッチを無効にすることはデータ取得での最も安全な方法ですが、行ごとのデータ取得はサーバのパフォーマンスを大幅に低下させます。[配列フェッチサイズ]オプションの値が大きくなると行を取得する速度が上がりますが、システムメモリが十分であることを確認してください。</p> <p>デフォルトの設定は 20 です。</p>
配列バインドサイズ	<p>リポジトリに送信する前に、Connection Server で使用されるバインド配列のサイズを入力します。バインド配列は、ロードされてリポジトリに送信されるデータのバッチを Connection Server が保存するメモリの領域です。バインド配列がいっぱいになると、データベースに送信されます。通常、バインド配列の許容量が大きければ、1 回の作業でより多くの行がロードされるため、パフォーマンスが向上します。</p>
ログインタイムアウト	<p>接続がタイムアウトになるまでの秒数を指定します。この時間を過ぎると、エラーメッセージが表示されます。</p>
所有者名	<p>DB2 接続の場合は、このパラメータはテーブルの所有者の名前をテーブル名の接頭語として追加し、テーブル命名に関する DB2 の規則に合わせます。</p>
テーブル接尾語	<p>DB2 接続の場合は、このパラメータはテーブル名に接尾語を追加し、テーブル命名に関する DB2 の規則に合わせます。</p>
ConnectInit	<p>DB2 接続の場合は、このパラメータによって、データベースに接続するときに一度だけ実行される SQL を追加できます。</p>
エイリアステーブル	<p>Essbase 接続の場合は、Essbase の中で定義したエイリアステーブルの名前です。</p>
メジャーディメンション	<p>Essbase 接続の場合は、メジャーを格納するために選択したディメンションです。</p>
文字列の最大サイズ	<p>テキストファイルや Excel ファイルなど、リレーショナルデータベースではないソースへの接続の場合は、このパラメータは、実際のサイズにかかわらず、すべての文字列に対して返されるサイズを定義します。デフォルト値は 1024 です。</p>

### 5.2.2.3 SAP HANA 接続のパラメータ

パラメータ	説明
認証モード	<ul style="list-style-type: none"> <li>指定したユーザ名とパスワードを使用する: ログインの詳細を認証として使用します。</li> <li>BusinessObjects 認証マップを使用する: レポートを最新表示するには、SAP BusinessObjects アカウントに関連付けられたデータベースユーザパスワードが要求されます。これは dbuser パラメータおよび dbpass パラメータを使用して設定します。これらは管理レベルで設定されます。このオプションの設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> </ul>
ユーザ名	[認証モード] が [指定したユーザ名とパスワードを使用する] の場合、データベースにアクセスするユーザ名。
パスワード	[認証モード] が [指定したユーザ名とパスワードを使用する] の場合、データベースにアクセスするパスワード。
サーバ: (<ホスト>:<ポート>)	<p>HANA サーバのホストおよびポートの詳細</p> <p><b>注</b> HANA インスタンス番号は、ポート番号の 2 番目および 3 番目の数で表されています。たとえば、ポート番号が 30215 の場合、HANA インスタンス番号は 02 になります。</p>

### 5.2.2.4 SAP NetWeaver BW データソースの接続パラメータ

パラメータ	説明
認証モード	<p>ユーザのログイン認証情報の認証に使用する方法。</p> <ul style="list-style-type: none"> <li>指定したユーザ名とパスワードを使用する: ログインの詳細を認証として使用します。</li> <li>SAP BusinessObjects 認証マップを使用する: レポートを最新表示するには、SAP BusinessObjects アカウントに関連付けられたデータベースユーザパスワードが要求されます。これは dbuser パラメータおよび dbpass パラメータを使用して設定します。これらは管理レベルで設定されます。このオプションの設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> <li>表示時のレポートの最新表示にシングルサインオンを使用する: このオプションを選択すると、CMS へのアクセスに使用したユーザ名とパスワードが、自動的にデータベースへのログインパラメータとして使用されます。シングルサインオン (SSO) の設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> </ul>
クライアント	SAP NetWeaver BW 上のクライアントの識別用に使用する番号。
ゲートウェイホスト	ネットワークゲートウェイが稼働しているホストの名前。
ゲートウェイサービス	ネットワークゲートウェイで稼働しているサービスの名前。
グループ名	<p>アプリケーションサーバのグループ名。SAP NetWeaver BW の負荷分散に使用されます。</p> <p>負荷分散の詳細については、SAP Java Connector のマニュアルを参照してください。</p>
インフォプロバイダ	データファンデーション内のスノーフレイクスキーマの中心のファクトテーブルとして使用される、SAP NetWeaver BW 上の InfoCube または MultiProvider の名前。
言語	SAP NetWeaver BW への接続に使用する言語コード。
言語の保存	<p>SAP NetWeaver BW への接続に使用する言語を指定します。</p> <p>[言語の保存] をオンにすると、[言語] フィールドの値が使用されます。</p> <p>[言語の保存] をオフにすると、ユーザのセッションからの値が使用されます。</p>

パラメータ	説明
メッセージサーバ名	メッセージサーバが稼働しているホスト名。SAP NetWeaver BW の負荷分散に使用されます。  負荷分散の詳細については、SAP Java Connector のマニュアルを参照してください。
パスワード	データフェデレーションアプリケーションで入力する、ユーザ名のパスワード
システム	SAP システムの名前。SAP NetWeaver BW の負荷分散に使用されます。  負荷分散の詳細については、SAP Java Connector のマニュアルを参照してください。
システム番号	接続する SAP NetWeaver BW システムを識別する番号。
使用可能であれば SNC を使用します	SNC を使用する場合、このチェックボックスをオンにしてください。
ユーザ名	データフェデレーションアプリケーションがデータのソースに接続するために使用するユーザ名

#### 5.2.2.4.1 InfoProvider ファクトテーブルの選択

SAP NetWeaver BW 接続を作成する場合、[InfoProvider ファクトテーブルを選択してください] ダイアログボックスでは、データファンデーション内のスノーフレイクスキーマの中心となるファクトテーブルを選択することができます。

フィルタボタンを使用すると、インフォプロバイダの種類でフィルタすることができます。

#### 5.2.2.5 SAS データソースの接続パラメータ

パラメータ	説明
認証モード	<p>ユーザのログイン認証情報の認証に使用する方法。</p> <ul style="list-style-type: none"> <li>指定したユーザ名とパスワードを使用する: ログインの詳細を認証として使用します。</li> <li>SAP BusinessObjects 認証マップを使用する: レポートを最新表示するには、SAP BusinessObjects アカウントに関連付けられたデータベースユーザパスワードが要求されます。これは dbuser パラメータおよび dbpass パラメータを使用して設定します。これらは管理レベルで設定されます。このオプションの設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> <li>表示時のレポートの最新表示にシングルサインオンを使用する: このオプションを選択すると、CMS へのアクセスに使用したユーザ名とパスワードが、自動的にデータベースへのログインパラメータとして使用されます。シングルサインオン (SSO) の設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> </ul>
パスワード	データフェデレーションアプリケーションで入力する、ユーザ名のパスワード
ポート	接続先のポート
ホスト名	SAS データベース用に SAS/SHARE が稼働しているサーバのホスト名
SAS/SHARE サーバに事前定義されていないデータセットの使用	このチェックボックスをオンにすると、SAS/SHARE サーバに事前定義されていない複数のデータセットにアクセスできます。これらのデータセットは現在の SAS 設定に含まれていません。詳細は、SAS/SHARE サーバに定義されていないデータセットの使用に関するドキュメントを参照してください。
ユーザ名	データフェデレーションアプリケーションがデータのソースに接続するために使用するユーザ名

#### 5.2.2.5.1 SAS/SHARE サーバに定義されていないデータセットの使用

SAS 接続を作成する場合、[SAS データセットの設定] ダイアログボックスでは、データファンデーションを設定して、SAS/SHARE サーバに定義されていない複数のデータセットにアクセスできます。これらのデータセットは現在の SAS 設定に含まれていません。

定義されていないデータセットを設定するには、以下の手順を使用します。

- 1 [SAS/SHARE サーバに事前定義されていないデータセットの使用] をオンにします。

- 2 [追加]をクリックして、[場所]フィールドに、使用しているオペレーティングシステムで要求された形式でデータセットへのパスを入力します。
- 3 [ライブラリ名]フィールドにデータセットへの参照に使用する名前を入力します。
- 4 必要に応じて [追加] をクリックし、その他のデータセットを追加します。
- 5 [完了] をクリックします。

### 5.2.3 ミドルウェアドライバの選択

ターゲットデータベース用の正しいミドルウェアバージョンに接続するために、接続ドライバを選択します。接続ドライバは、ミドルウェア内の情報を SAP BusinessObjects アプリケーションのユーザインタフェースにマップする SAP BusinessObjects ドライバです。

ターゲットデータベースのデータベースとミドルウェアノードを展開し、接続ドライバを選択します。[次へ] をクリックしてウィザードを続行します。

#### 注

SAP NetWeaver BW および SAS 接続は、リポジトリ内で直接接続を作成している場合のみリストされます。

#### 関連項目

- ・ 54 ページの [データソースパラメータの設定](#)

## 5.3 OLAP 接続の作成

新しい OLAP 接続ウィザードを使用して、OLAP データソースへのローカル接続およびセキュリティ接続を作成します。

[インフォメーションデザインツール] でローカル接続を作成する前に、[ローカルプロジェクト] ビューにプロジェクトがある必要があります。ローカルプロジェクトの作成の詳細については、関連項目を参照してください。

#### 注

インフォメーションデザインツールで作成する OLAP 接続は、ユニバースデザインツール ではサポートされていません。また、ユニバースデザインツールで作成された OLAP 接続は、インフォメーションデザインツールでユニバースを構築するためには使用できません。

- 1 次のいずれかを実行します。
  - ・ ローカル接続を作成するには、[ローカルプロジェクト] ビューからプロジェクトフォルダを選択します。[ファイル] > [新規] > [OLAP 接続] を選択します。
  - ・ セキュリティ接続を作成するには、[リポジトリリソース] ビューで、セキュリティ接続を作成するリポジトリでセッションを開きます。リポジトリ内の接続フォルダまたはサブフォルダを右クリックし、[OLAP 接続の挿入] を選択します。

2 新規 OLAP 接続ウィザードの手順に従って、以下の情報を入力します。

- ・ 接続名
- ・ ターゲットデータベースのミドルウェアのドライバ
- ・ OLAP データベースに接続するための認証パラメータ
- ・ 接続先の OLAP キューブ

個々の手順で支援が必要な場合は、ウィザードダイアログボックスでヘルプアイコンをクリックしてください。

#### 関連項目

- ・ 24 ページの[ローカルプロジェクトの作成](#)」
- ・ 63 ページの[OLAP ミドルウェアドライバの選択](#)」

### 5.3.1 OLAP ミドルウェアドライバの選択

この節では、新しい OLAP 接続ウィザードの OLAP ドライバ選択ページについて説明します。

OLAP サーバに接続するには、OLAP ドライバを選択します。OLAP ドライバは、OLAP サーバミドルウェアの情報を SAP BusinessObjects アプリケーションのユーザインタフェースにマップします。

ターゲット OLAP サーバによって、ミドルウェアノードを展開し、ターゲットドライバを選択します。

[次へ] をクリックしてウィザードを続行します。

#### 関連項目

- ・ 52 ページの[SAP NetWeaver BW 接続について](#)」

### 5.3.2 OLAP データソースのログインパラメータの設定

接続パラメータは、接続を定義しているデータソースの種類によって異なります。接続パラメータの詳細については、関連トピックのからリンクを選択してください。

#### 関連項目

- ・ 64 ページの[SAP BW OLAP 接続のログインパラメータ](#)」
- ・ 65 ページの[MSAS OLAP 接続のログインパラメータ](#)」

### 5.3.2.1 SAP BW OLAP 接続のログインパラメータ

[新しい接続] ウィザードの [ログインパラメータ] ダイアログボックスには、次のパラメータが含まれています。

ログインパラメータ	説明
認証モード	<ul style="list-style-type: none"> <li>指定したユーザ名とパスワードを使用する: ログインの詳細を認証として使用します。</li> <li>BusinessObjects 認証マップを使用する: レポートを最新表示するには、BusinessObjects アカウントに関連付けられたデータベースユーザパスワードが要求されます。これは dbuser パラメータおよび dbpass パラメータを使用して設定します。これらは管理レベルで設定されます。このオプションの設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> <li>表示時のレポートの最新表示にシングルサインオンを使用する: このオプションを選択すると、CMS へのアクセスに使用したユーザ名とパスワードが、自動的にデータベースへのログインパラメータとして使用されます。シングルサインオン (SSO) の設定については、『Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> </ul>
使用可能であれば SNC を使用します	SNC を使用する場合、このチェックボックスをオンにしてください。
クライアント	SAP NetWeaver BW システムのクライアントの識別に使用する番号 (必須)。
ユーザ名	[認証モード] が [指定したユーザ名とパスワードを使用する] の場合、OLAP サーバへのアクセスに使用するユーザ名。
パスワード	[認証モード] が [指定したユーザ名とパスワードを使用する] の場合、OLAP サーバへのアクセスに使用するパスワード。
言語	<p>接続に使用する言語。</p> <p><b>注</b> 接続言語は、ユニバースを生成する言語を決定します。</p>



ログインパラメータ	説明
言語の保存	次のうち接続に使用する言語。 <ul style="list-style-type: none"> <li>・ [言語の保存] をオンにすると、[言語] フィールドの値が使用されます。</li> <li>・ [言語の保存] をオフにすると、ユーザのセッションからの値が使用されます。</li> </ul>
ログインモードまたはサーバの種類	負荷分散を使用せずに SAP サーバに直接接続するには、[アプリケーションサーバ] を選択します。  SAP 負荷分散機能の利点を活用するには、[メッセージサーバ] を選択します。
アプリケーションサーバ	SAP アプリケーションサーバの名前または IP アドレスを選択するか、入力します。
システム番号	システム番号を入力します (たとえば、00)。アプリケーションサーバのログインモードには必須です。
システム ID	<p>[メッセージサーバ] ログインモードを使用する場合は、[メッセージサーバ]、[ログオングループ]、および [システム ID] を入力します。</p> <p><b>注</b></p> <p>メッセージサーバに適切に接続するためには、メッセージサーバのシステム ID を、このアプリケーションをホストするマシンの以下のファイルに追加する必要があります。</p> <p>C:\WINDOWS\system32\drivers\etc\services</p> <p>既存のファイルの最後に以下の行を追加します。</p> <p>sapmsXXX &lt;tab&gt; 3601/tcp</p> <p>sapms は SAP メッセージサーバ、xxx は使用されているサーバのシステム ID、3601/tcp はデフォルトの通信用 TCP ポートを表します。</p>
ログオングループ	
メッセージサーバ	

### 5.3.2.2 MSAS OLAP 接続のログインパラメータ

[新しい接続] ウィザードの [ログインパラメータ] ダイアログボックスには、次のパラメータが含まれています。

ログインパラメータ	説明
認証モード	<ul style="list-style-type: none"> <li>指定したユーザ名とパスワードを使用する: ログインの詳細を認証として使用します。</li> <li>BusinessObjects 認証マップを使用する: レポートを最新表示するには、BusinessObjects アカウントに関連付けられたデータベースユーザパスワードが要求されます。これは dbuser パラメータおよび dbpass パラメータを使用して設定します。これらは管理レベルで設定されます。このオプションの設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> <li>表示時のレポートの最新表示にシングルサインオンを使用する: このオプションを選択すると、CMS へのアクセスに使用したユーザ名とパスワードが、自動的にデータベースへのログインパラメータとして使用されます。シングルサインオン (SSO) の設定については、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。</li> </ul>
サーバ	<p>次のいずれかを入力します。</p> <ul style="list-style-type: none"> <li>MSAS サーバで公開して設定した MSAS ライブラリの URL</li> <li>MSAS データソースのサーバ名</li> <li>MSAS キューブファイルのフルパスファイル名。次のように、二重引用符で囲んだフルパスファイル名を入力します。</li> </ul> <p>"Z:¥All cubes¥test.cub"</p> <p><b>注</b> キューブファイルが SAP BusinessObjects ホスト以外のホストシステムに配置されている場合、SAP BusinessObjects ホストとそのホストは接続を共有している必要があります。キューブファイルへの接続は、SAP BusinessObjects ホスト上で直接作成する必要があります。</p>
ユーザ名	[認証モード] が [指定したユーザ名とパスワードを使用する] の場合、OLAP サーバへのアクセスに使用するユーザ名。
パスワード	[認証モード] が [指定したユーザ名とパスワードを使用する] の場合、OLAP サーバへのアクセスに使用するパスワード。
言語	接続に使用する言語。

### 5.3.3 OLAP キューブの選択

この節では、新しい OLAP 接続ウィザードのキューブの選択のページについて説明します。

キューブを指定せずに接続を作成するには、[接続内でキューブを指定しない] オプションを選択します。この場合、ユーザが接続にアクセスするたびに、ビジネスレイヤを作成するときか、またはクエリツールおよびレポートツールでキューブの選択を求められます。

常にキューブを接続に関連付けるには、[接続内でキューブを指定する] オプションを選択します。

[キューブの選択] ページには、ターゲットデータベースで使用できるキューブの一覧が表示されます。[検索] テキストボックスに、検索文字列を入力します。一覧からキューブを選択します。

#### 注

[SAP BICS Client] ドライバで、[接続内でキューブを指定する] オプションを選択します。[Information Area] フォルダで、インフォキューブを選択します。次に、接続の BEx クエリを選択します。

[完了] をクリックしてウィザードを終了します。

### 5.4 接続ショートカットの作成

接続を公開する場合、[ローカルプロジェクト] ビュー内に接続ショートカットの作成オプションがあります。次の手順を使用して、既存のセキュリティ接続の接続ショートカットを作成します。

[ローカルプロジェクト] ビュー内にローカルプロジェクトがある必要があります。

- 1 [リポジトリリソース] ビューで、セキュリティ接続が保存されているリポジトリでセッションを開きます。
- 2 [接続] フォルダまたはサブフォルダで、接続名を右クリックします。
  - ・ OLAP 接続の場合、[OLAP 接続のショートカットの作成] を選択します。
  - ・ リレーショナル接続の場合、[リレーショナル接続のショートカットの作成] を選択します。
- 3 [ローカルプロジェクトの選択] ダイアログボックスで、中にショートカットを作成するオブジェクトを選択します。

#### 関連項目

- ・ 24 ページの [ローカルプロジェクトの作成](#)

### 5.5 接続および接続ショートカットの編集について

コネクションエディタを使って接続および接続のショートカットを編集できます。

ローカル接続または接続のショートカットを編集するには、[ローカルプロジェクト] ビューで接続またはショートカット名をダブルクリックします。

リポジトリリソースビューでセキュリティ保護された接続を編集するには、接続が公開されるリポジトリでセッションを開きます。接続フォルダまたはサブフォルダで、接続名をダブルクリックします。

コネクションエディタでは、接続の種類に応じて以下のタスクを実行できます。

#### リレーショナル接続

[一般情報] 枠で以下の手順を実行します。

- ・ 接続パラメータを編集するには [編集] をクリックします。
- ・ ミドルウェアドライバを変更するには、[ドライバの変更] をクリックします。
- ・ データベースサーバの可用性をテストするには、[接続のテスト] をクリックします。

[接続定義] 枠には接続について保存されている情報が表示されます。

[値の表示] 枠では、接続によって参照されるテーブルの値を参照できます。

#### OLAP 接続

[一般情報] 枠で以下の手順を実行します。

- ・ 接続ログインパラメータとキューブ選択を編集するには、[編集] をクリックします。
- ・ キューブサーバの可用性をテストするには、[接続のテスト] をクリックします。

[接続定義] 枠には接続について保存されている情報が表示されます。

[値の表示] 枠では、キューブ内のオブジェクトとそのプロパティを参照できます。

[クエリ] 枠では、キューブオブジェクトをドラッグアンドドロップして、MDX クエリを作成し、キューブにクエリを実行できます。

#### 接続ショートカット

[一般情報] 枠で以下の手順を実行します。

- ・ ショートカットが参照するセキュリティ保護された接続を変更するには、[接続の変更] をクリックします。変更する接続が公開されるリポジトリにセッションを開きます。
- ・ データベースまたはキューブサーバの可用性をテストするには、[接続のテスト] をクリックします。

## データファンデーションを操作する

### 6.1 データファンデーションの概要

データファンデーションには、関連のあるテーブルのスキームと、1 つ以上のビジネスレイヤの基礎として使用される 1 つ以上のリレーショナルデータベースからの結合が含まれます。

データファンデーションのリレーショナル接続を参照します。接続で参照されたデータベースからテーブルと結合を挿入します。

データファンデーションエディタを使用して、派生テーブル、エイリアステーブル、計算列、追加の結合、コンテキスト、プロンプト、および値の一覧を追加することにより、データファンデーションを強化できます。利用できる機能は、データファンデーションの種類に依存します。データファンデーションの種類の詳細については、関連項目を参照してください。

同じデータファンデーションに任意の数のビジネスレイヤを構築できます。この場合、データファンデーションは、複数のユニバースの基になります。

#### 関連項目

- ・ 69 ページの[データファンデーションの種類について](#)
- ・ 71 ページの[データファンデーションの作成方法](#)

#### 6.1.1 データファンデーションの種類について

シングルソースおよび複数ソース有効は、2 種類のデータファンデーションにより、異なるデータファンデーションの機能を利用することができます。この関連トピックでは、用途に適した種類を選択できるようにする情報について説明します。



#### 関連項目

- ・ 73 ページの[シングルソースのデータファンデーションについて](#)
- ・ 73 ページの[複数のソースが有効化されたデータファンデーションについて](#)

## 6.2 データファンデーションエディタについて

データファンデーションエディタは、データファンデーションビューと参照用の枠に分割されています。

データファンデーションビューでは、テーブルや結合がグラフィックで表されます。[マスタ]ビューには、すべてのテーブルと結合が含まれ、削除できません。テーブルのサブセットを含むカスタムビューを定義できます。ビュー枠の下部にあるタブを使ってビューにアクセスします。カスタムビューの詳細については、関連項目を参照してください。

データファンデーションビューでは、[挿入]  および [検出]  メニューのコマンドを使って、またはビューから直接オブジェクトをクリックしてテーブルや結合に作業できます。

参照枠では、データファンデーションの異なる要素に作業できます。対応するタブをクリックして枠にアクセスします。

- ・ 接続
- ・ データファンデーション (テーブルおよび結合のツリービューを表示)
- ・ エイリアスとコンテキスト
- ・ パラメータおよび値の一覧
- ・ プロパティ

各参照枠での操作の詳細については、関連項目を参照してください。


### データファンデーションビューのナビゲート

テーブルに使用できるコマンドメニューにアクセスするには、データファンデーションビューのテーブルヘッダを右クリックします。複数のテーブルを選択するには、CTRL キーを押しながらテーブルヘッダをクリックします。

列に使用できるコマンドにアクセスするには、データファンデーションビューでテーブルの列名を右クリックします。

テーブルのコマンドでは、右クリックメニューを使用して、データファンデーションで関連するテーブルを特定するために使用できます。

- ・ [関連テーブルの選択] では、選択したテーブルに結合によってつながられているすべてのテーブルが自動的に選択されます。
- ・ [関連テーブルを強調表示] では、選択されたテーブルに結合によってつながられていないテーブルがグレイで表示されます。
- ・ [エイリアスを強調表示] では、選択された標準テーブルとそのエイリアステーブルを除いてすべてのテーブルがグレイで表示されます。
- ・ [元のテーブルを強調表示] では、選択したエイリアステーブルとその基になっている標準テーブル以外のすべてのテーブルがグレイで表示されます。
- ・ [選択時に中央へ] によりデータファンデーション表示上でズームを一時的に変更できるため、選択項目のすべてのテーブルは表示ウィンドウに表示されます。

検索パネルを使用してデータファンデーションを詳細検索することもできます。検索パネルを開くには、をクリックします。

#### 関連項目

- ・ 95 ページの[カスタムデータファンデーションビューについて](#)
- ・ 76 ページの[データファンデーションの接続について](#)
- ・ 90 ページの[コンテキストについて](#)
- ・ 93 ページの[データファンデーション内のパラメータと値の一覧について](#)
- ・ 93 ページの[データファンデーションのプロパティについて](#)
- ・ 96 ページの[データファンデーションでのテーブルおよび列の検索](#)
- ・ 97 ページの[選択時のビューの中央揃え](#)

## 6.3 データファンデーションの作成方法

この項目では、インフォメーションデザインツールでデータファンデーションを作成する手順の概要を説明します。詳細情報については、この項目の最後にある関連項目を参照してください。

- 1 データを提供するビジネスレイヤのビジネス要件を定義して、データファンデーションのデザインを準備します。
- 2 データファンデーションの種類が、単一ソースか複数ソースを有効化する必要があるのかを決定します。データファンデーションの種類に従い、SQL 構造を定義するのに使用する SQL 構文と同様に、接続の種類と接続数が利用可能です。
- 3 データファンデーションを作成するローカルプロジェクトを選択するか定義します。
- 4 データファンデーションを定義するために必要なデータベースを参照するリレーショナル接続を、選択または定義します。データファンデーション用に選択されたローカルプロジェクトに、ローカル接続を作成します。セキュリティ接続を参照するか、または複数ソース有効データファンデーションを作成する場合、ローカルプロジェクトに接続ショートカットを作成します。
- 5 [新しいデータファンデーション]ウィザードを使用して、データファンデーションを作成します。ウィザードでは、データファンデーションの名前と説明を入力して、接続を選択します。複数ソースが有効化されたデータファンデーションでは、データファンデーションの表示内で各テーブルの基になるデータベースを区別できるように、各接続のプロパティを定義します。

[新しいデータファンデーション]ウィザードを開始するには、[ファイル] > [新規] > [データファンデーション]を選択します。

- 6 データファンデーションが作成されたら、データファンデーションエディタを使用して、接続内で参照されるデータベースから必要なテーブルや結合を挿入します。[テーブルの挿入] コマンドを使用して、一度に複数のテーブルを挿入し、自動的に検出され挿入されるキー、結合、カーディナリティおよび行数を要求します。SAP NetWeaver BW 接続では、自動的にテーブルと結合が挿入されます。また、テーブルを挿入した後で、キー、結合、およびカーディナリティを検出するために、[検出]メニューのコマンドを使用することもできます。

#### 注

異なる接続から参照されているテーブル間の結合を検出するには、[結合の検出] コマンドを使用します。

- 7 データファンデーションエディタでは、データファンデーションの機能をいくつかの方法で拡張できます。たとえば次のとおりです。
  - ・ 別の接続によって参照されるデータベースからテーブルをリンクする結合を含めて、新しい結合を挿入し、既存の結合を編集します。
  - ・ 計算列の挿入
  - ・ 派生テーブルの挿入
  - ・ オプションのプロンプトを使用したパラメータの挿入
  - ・ プロンプトに関連付ける値の一覧の挿入
- 8 結合のパスを確認し、カーディナリティ、エイリアステーブル、およびコンテキストを検出して、ループなどを解決します。[エイリアスとコンテキスト] 枠でコマンドを使用し、エイリアスとコンテキストを自動的に検出します。
- 9 データファンデーションを完了する場合、[整合性のチェック] コマンドを使用して、データファンデーション内のテーブル、列、および結合を確認します。  
整合性のチェックを実行するには、[データファンデーション] 枠内でデータファンデーション名を右クリックし、[整合性のチェック] を選択します。
- 10 インフォメーションデザインツール内の別のリソース同様、データファンデーションは共有することができます。これにより、別のデザイナーがこのデータファンデーションを使用して作業を行うことができます。プロジェクト同期ビューを使用して、共有リソースを使用するローカルデータファンデーションを同期します。
- 11 データファンデーションエディタでデータファンデーションを保持します。  
データファンデーションの保持に役立つコマンドは、次の一覧のとおりです。
  - ・ テーブルまたは列を変更する場合、[ローカルの依存関係の表示] を使用して、変更によって影響を受ける可能性のあるビジネスレイヤおよびオブジェクトを見つけることができます。
  - ・ 構造を最新表示すると、接続内で参照されるデータベースに発生した変更で、データファンデーションを更新します。
  - ・ 接続の追加または変更
  - ・ データファンデーションプロパティ内の SQL 生成パラメータの設定

#### 関連項目

- ・ 69 ページの[データファンデーションの種類について](#)
- ・ 24 ページの[ローカルプロジェクトの作成](#)
- ・ 53 ページの[リレーショナル接続の作成](#)
- ・ 67 ページの[接続ショートカットの作成](#)
- ・ 79 ページの[データファンデーションへのテーブルの挿入](#)
- ・ 75 ページの[NetWeaver BW 接続を使ったデータファンデーション](#)
- ・ 86 ページの[結合の検出について](#)
- ・ 91 ページの[ループの解決](#)
- ・ 27 ページの[プロジェクト同期について](#)
- ・ 94 ページの[ローカル依存関係の表示](#)
- ・ 94 ページの[データファンデーションの最新表示について](#)
- ・ 76 ページの[データファンデーションの接続について](#)
- ・ 93 ページの[データファンデーションのプロパティについて](#)



## 6.4 シングルソースのデータファンデーションについて

シングルソースのデータファンデーションは 1 つの接続をサポートします。ローカル接続またはセキュリティ接続が可能です。つまり、ローカルまたはリポジトリのいずれかにデータファンデーションに基づくユニバースを公開することができます。

シングルソースのデータファンデーションは、派生テーブル、計算列、および結合式でのデータベース固有の SQL 構文をサポートします。データベース固有の SQL 構文を使用することで、標準の SQL-92 ではなく特定のデータベースで提供される関数や演算子を使用できます (たとえば Oracle の分析関数)。このデータファンデーションに基づくユニバースをローカルフォルダに公開する場合は、単一のソースを選択する必要があります。

シングルソースのデータファンデーションは、次のような場合に推奨されます。

- ・ データベース固有の SQL 構文のみを使用します。
- ・ ユニバースをローカルで公開し、リポジトリの外で作業します。

### 関連項目

- ・ 73 ページの[複数のソースが有効化されたデータファンデーションについて](#)
- ・ 71 ページの[データファンデーションの作成方法](#)

## 6.5 複数のソースが有効化されたデータファンデーションについて

複数のソースが有効化されたデータファンデーションは、1 つまたは複数の接続をサポートします。接続は、データファンデーションを作成するときや、後でいつでも追加できます。複数のソースが有効化されたデータファンデーションでは、セキュリティ保護された接続のみがサポートされ、このタイプのデータファンデーションを基にしたユニバースはリポジトリにのみ公開できます。

サポートされている追加のリレーショナル接続は、次のとおりです。

- ・ SAP NetWeaver BW 接続
- ・ SAS 接続

計算列、派生テーブル、および結合式では、SQL-92 標準構文がデフォルトになっています。また、SAP BusinessObjects SQL 関数が使用できます。複数のソースが有効化されたデータファンデーションでは、データベース固有の派生テーブルまたは計算列を定義することで、データベース固有の SQL 構文を使用できます。データベース固有の SQL 構文を使用することで、標準の SQL-92 ではなく特定のデータベースで提供される関数や演算子を使用できます (たとえば Oracle の分析関数)。

次のような場合は、複数のソースが有効化されたデータファンデーションが必要です。

- ・ 複数のリレーショナルデータソースからテーブルや結合を挿入します。
- ・ SAP NetWeaver BW または SAS 接続を使用します。

- ・ SQL-92 標準構文および SAP BusinessObjects SQL 関数を使用する場合。

これらの状況の詳細については、関連項目を参照してください。

#### 関連項目

- ・ 74 ページの[複数の接続があるデータファンデーション](#)
- ・ 75 ページの[NetWeaver BW 接続を使ったデータファンデーション](#)
- ・ 75 ページの[複数のソースが有効化されたデータファンデーションの SQL 式](#)
- ・ 73 ページの[シングルソースのデータファンデーションについて](#)
- ・ 71 ページの[データファンデーションの作成方法](#)

### 6.5.1 複数の接続があるデータファンデーション

データファンデーションに複数の接続を追加するには、データファンデーションを作成する時に複数のソースを有効化した種類を選択する必要があります。

データファンデーションを作成するときに複数の接続を選択できます。複数のソースが有効化されている既存のデータファンデーションにも接続を追加できます。コネクションはセキュリティ保護されている必要があるため、リポジトリで使用する必要があります。接続は、ローカルプロジェクトの接続ショートカットで表されます。

複数のソースを有効化したデータファンデーションの接続には、以下の追加のプロパティがあります。

- ・ データファンデーション内の接続を識別するため、および SQL 式内のテーブル名を変更するために使用されるショート名。接続を追加するときにショート名を指定します。この名前はデータファンデーション内で一意である必要があり、40 文字に制限されています。接続のショート名を変更すると、SQL 式は新しい名前ですべて自動的に更新されます。
- ・ 接続の色。この色は、データファンデーションビューのテーブルヘッダに使用されます。接続を追加するときに色を選択します。接続の色はいつでも変更できます。
- ・ クエリサーバへの接続を識別するために使用されるカタログ。デフォルトのカタログ名は、複数のソースが有効化されたデータファンデーションにはじめて接続が追加されたときにクエリサーバに自動的に登録されます。
- ・ SAP NetWeaver BW 接続の場合の、テーブルと結合の自動挿入に関するプロパティです。これらのプロパティの詳細については、関連項目を参照してください。

複数のソースが有効化されたデータファンデーションでは、SQL 式の中のテーブル名は、<@catalog(short name). "database\_qualifier.database\_owner".table\_name> の形式を取ります。

複数のソースの結合は、異なる接続からのテーブル間で作成されます。異なる接続で参照されるテーブル間の結合を検出するには、[結合の検出] コマンドを使用するか、[結合の挿入] コマンドで明示的に定義します。

#### 関連項目

- ・ 75 ページの[NetWeaver BW 接続を使ったデータファンデーション](#)
- ・ 76 ページの[データファンデーションの接続について](#)

## 6.5.2 複数のソースが有効化されたデータファンデーションの SQL 式

複数のソースが有効化されたデータファンデーションに結合、計算列、派生テーブルを定義する SQL 式では、SQL-92 ANSI 標準構文が使用されます。

SQL-92 式では、SAP BusinessObjects SQL 関数および @ 関数を含められます。含められる @ 関数は、式の種類に依存します。詳細については、関連項目を参照してください。

SQL-92 ではなくデータベースが提供する関数または演算子 (Oracle 分析関数など) を使用するには、データベース固有の計算列と派生テーブルを定義します。SQL 式エディタのチェックボックスを使用して、データベース固有の SQL を使用できます。

データベース固有の計算列および派生テーブルでは、関連する接続の SQL 構文がサポートされています。以下のルールが、データベース固有の SQL 式に適用されます。

- ・ 1 つの接続では、標準テーブルとデータベース固有の派生テーブルのみを参照できます。
- ・ SAS または SAP NetWeaver BW 接続ではテーブルは参照できません。
- ・ 特定の制限付きで @ 関数を含められます。詳細については、関連項目を参照してください。

### 関連項目

- ・ 225 ページの[SQL 関数リファレンス](#)」
- ・ 205 ページの[関数のリファレンス](#)」

## 6.5.3 NetWeaver BW 接続を使ったデータファンデーション

SAP NetWeaver BW 接続を複数のソースを有効にしたデータファンデーションに追加すると、テーブルと結合が自動的に挿入されます。

自動挿入では以下が実行されます。

- ・ SAP NetWeaver BW 接続パラメータに指定されている InfoProvider を基に、ファクトテーブルとそれに関連するマスタデータおよびテキストテーブルを挿入します。
- ・ 各テーブルタイプにファミリーを作成します。
  - ・ ファクトテーブル名にプレフィックスとして文字 I (InfoCube) を付け、ファミリー [InfoProvider ファクトテーブル] に割り当てます。
  - ・ マスタテーブルにプレフィックスとして文字 D (Dimension) を付け、ファミリー [ディメンションテーブル] に割り当てます。
  - ・ テキストテーブルにプレフィックスとして文字 T (Text) を付け、ファミリー [テキストテーブル] に割り当てます。
- ・ すべてのディメンションおよびテキストテーブルにエイリアステーブルを作成します。
- ・ テーブルキーを検出して挿入します。

- ・ 結合を検出して挿入します。
- ・ 時間に依存するデータを処理するために、必要に応じてテーブルに入力列を作成します。キー日付と呼ばれるパラメータをデータファンデーションに作成します。デフォルトでは、クエリ実行時にキー日付パラメータは入力を要求されません。これは現在の日付に自動的に割り当てられます。


#### 注

自動挿入をオフにするには、接続を追加するときに接続の詳細プロパティの [テーブルの検出] チェックボックスの選択を外します。

#### 関連項目

- ・ 52 ページの [SAP NetWeaver BW 接続について](#)

## 6.6 データファンデーションの接続について

データファンデーションの接続は、データファンデーションエディタの [接続] 枠に一覧表示されます。各接続で参照されている表が一覧表示されます。デフォルトでは、すべての修飾子と所有者の表が一覧表示されます。現在使用されている修飾子/所有者のみを表示するには、[修飾子および所有者の表示] アイコン  をクリックします。

#### 接続の操作

[接続] 枠から、接続に以下の操作を実行できます。データファンデーションエディタを開くには、[ローカルプロジェクト] ビューにあるデータファンデーション名をダブルクリックします。データファンデーションエディタで、[接続] タブをクリックします。


- ・ [変更] では、接続と、関連するプロパティを変更できます。このタスクの詳細については、関連項目を参照してください。
- ・ [接続の追加] (複数のソースを有効にしたデータファンデーションのみ) では、データファンデーションに接続を追加できます。このタスクの詳細については、関連項目を参照してください。
- ・ [編集] では、コネクションエディタで接続、または接続ショートカットのプロパティが開きます。
- ・ [テスト] では、接続が参照しているデータベースが使用できるかをテストします。
- ・ [削除] (複数のソースを有効にしたデータファンデーションのみ) では、データファンデーションから接続を削除できます。接続自体はリポジトリに残り、カタログ名で登録されます。

#### 関連項目

- ・ 77 ページの [データファンデーション内での接続の変更](#)
- ・ 77 ページの [データファンデーションに対する接続の追加](#)
- ・ 78 ページの [データファンデーションのテーブルについて](#)

### 6.6.1 データファンデーションに対する接続の追加

データファンデーションに接続を追加するには、次の条件が必要です。

- ・ データファンデーションの種類が、複数ソース有効である必要があります。
  - ・ 追加する接続はリレーショナルで、セキュリティ接続である必要があります。
  - ・ 追加する各接続に対して、データファンデーションが保存されるローカルプロジェクト内に、接続ショートカットを作成する必要があります。
- 1 ローカルプロジェクト内でデータファンデーション名をダブルクリックし、データファンデーションエディタを開きます。
  - 2 データファンデーションエディタで、[接続] タブをクリックします。
  - 3 [接続] 枠で、[接続の追加] アイコン  をクリックします。  
[接続の追加] ダイアログボックスに、現在データファンデーション内に定義されている接続も含めて、使用できる接続が一覧表示されます。
  - 4 追加する各接続の接続ショートカット名の隣にあるチェックボックスを選択し、[次へ] をクリックします。
  - 5 [接続プロパティ] ダイアログボックスで、追加の接続プロパティを定義することができます。各接続が追加されるたびに、ダイアログボックスが開きます。  
複数ソース有効接続プロパティの詳細については、関連項目を参照してください。
  - 6 追加接続のプロパティの定義が完了したら、[完了] をクリックします。

#### 関連項目


- ・ 67 ページの[接続ショートカットの作成](#)
- ・ 74 ページの[複数の接続があるデータファンデーション](#)

### 6.6.2 データファンデーション内での接続の変更

変更する接続が、リレーショナル接続である必要があります。複数ソース有効データファンデーションの場合、接続にセキュリティが設定されている必要があります。

接続を変更する前に、データファンデーションが保存されるローカルプロジェクト内に、ローカル接続または接続ショートカットを作成する必要があります。

- 1 ローカルプロジェクト内でデータファンデーション名をダブルクリックし、データファンデーションエディタを開きます。
- 2 データファンデーションエディタで、[接続] タブをクリックします。
- 3 [接続] 枠で、変更する接続を選択します。接続を右クリックして、[変更...] を選択します。
- 4 新しい接続を選択します。変更方法は、データファンデーションの種類に依存します。

- ・ データファンデーションがシングルソースの場合、[接続の変更] ダイアログボックスには、現在定義されている接続を含め、使用可能な接続が一覧表示されます。変更する接続の隣にあるチェックボックスを選択し、[OK] をクリックします。
- ・ データファンデーションが複数ソース有効の場合、[接続の変更] ダイアログボックスには、現在定義されている接続に対する接続プロパティが表示されます。[接続] テキストボックスで  アイコンをクリックします。ダイアログボックスに、使用できる接続が一覧表示されます。変更する接続を選択し、[完了] をクリックします。

## 6.7 データファンデーションのテーブルについて

標準テーブルは、物理的なデータベーステーブルをデータファンデーションでグラフィックで表現したものです。標準テーブルは、データベーステーブルをデータファンデーションに挿入するときに作成します。

異なる所有者の複数のデータベース（修飾子と呼ばれます）が使用できる接続もあります。標準テーブルとその列は、データベースから完全修飾名を継承します。単一のソースを持つ標準テーブル名の構文は、`<database_qualifier.database_owner.table_name>` です。

### 注

修飾子と所有者が接続に関係ない場合もあり、この場合はテーブル名のみが使用されます。

複数のソースが有効化されたデータファンデーションも、接続をテーブル名で特定します。複数のソースが有効化された標準テーブル名の構文は、`<@catalog(short_name).database_qualifier.database_owner.table_name>` です。複数の接続があるデータファンデーションの詳細については、関連項目を参照してください。

標準テーブルを挿入したら、テーブルを編集して説明を追加し、列を変更できます。


テーブルの修飾子または所有者を変更するには、データファンデーションビューでテーブルヘッダを右クリックし、[修飾子/所有者の変更] を選択します。


データファンデーションのテーブルは、派生テーブルまたはエイリアステーブルの場合もあります。詳細については、関連項目を参照してください。

### 関連項目

- ・ 74 ページの [複数の接続があるデータファンデーション](#)
- ・ 88 ページの [派生テーブルについて](#)
- ・ 79 ページの [データファンデーションへのテーブルの挿入](#)
- ・ 80 ページの [データファンデーションテーブルの編集](#)
- ・ 80 ページの [テーブル名の大文字/小文字の設定](#)
- ・ 90 ページの [エイリアステーブルについて](#)

### 6.7.1 データファンデーションへのテーブルの挿入

データファンデーションにテーブルを挿入するには、データファンデーションビューの [挿入]  メニューから [テーブルの挿入] を選択します。

[テーブルの挿入] ダイアログボックスには、データファンデーションで定義された接続が表示されます。接続で参照されるデータベーステーブルを表示するには、接続を展開します。デフォルトでは、すべての修飾子と所有者の表が一覧表示されます。現在使用されている修飾子/所有者のみを表示するには、[修飾子および所有者の表示] アイコン  をクリックします。

テーブル名の隣にあるチェックボックスにチェックを付け、そのテーブルと、その中のすべての列をデータファンデーションに挿入します。

すでにデータファンデーションに挿入されているテーブルには、緑色のチェックマークの付いたアイコンが表示されます。既存のテーブルを挿入する場合は、エイリアステーブルが挿入され、エイリアステーブル名を入力するよう要求されます。

テーブルに値を表示するには、テーブル名を右クリックして、[テーブル値の表示] を選択します。1 つの列に値を表示するには、テーブルを展開し、列名を右クリックして [列の値の表示] を選択します。

テーブルを挿入したときに自動的に検出されデータファンデーションに挿入されるオブジェクトを選択します。

- ・ [キーの検出] は、データファンデーションのキー列を、データベーステーブルにあるものと同じように設定します。
- ・ [行数の検出] は、データファンデーションの各テーブルの行数を保存します。
- ・ [結合の検出] は、挿入されたテーブル間に結合を挿入します。

#### 注

複数のソースが有効化されたデータファンデーションでは、同じ接続によって参照されているテーブル間の結合のみが検出されます。異なる接続によって参照されているテーブル間の結合を検出するには、テーブルを挿入した後、[結合の検出] コマンドを使用します。

- ・ [カーディナリティの検出] は、データベース結合にあるものと同じように、結合のカーディナリティを保存します。

推奨される検出オプションはデフォルトで選択されています。デフォルトを変更するには、アプリケーションの基本設定ページを使用します。メインメニューから、[ウィンドウ] > [基本設定] > [インフォメーションデザインツール] > [データファンデーションエディタ] > [自動検出] を選択します。

#### ヒント

データファンデーションにテーブルを挿入するには、テーブルを [接続] 枠からドラッグして、データファンデーションビューにドロップします。

#### 関連項目

- ・ 86 ページの [結合の検出について](#)



## 6.7.2 データファンデーションテーブルの編集

データファンデーションのテーブルを編集するには、データファンデーションビューでテーブルヘッダを右クリックして、[編集] を選択します。

派生テーブルの場合は、[派生テーブルの編集] ダイアログボックスが表示されます。関連項目を参照してください。

エイリアスおよび標準テーブルの場合は、[テーブルの編集] ダイアログボックスでテーブル名および説明を編集できます。テーブル名はデータファンデーション内で一意である必要があります。

### 注

標準テーブル名を変更すると、データベーステーブルとのリンクが壊れます。エイリアスを使ったテーブル名の変更については、関連項目をご覧ください。

標準テーブルの場合は、リストボックスで新しいデータ型を選択することでも列のデータ型を変更できます。

### 注

データファンデーション構造を次回最新表示したときに、データベース内の列の元のデータ型が提案されます。

標準テーブルの場合は、リストボックスでキーのタイプを選択することで、キーを割り当てまたは割り当て解除できます。テーブルキーの詳細については、関連項目を参照してください。

### 関連項目

- ・ 88 ページの[派生テーブルについて](#)
- ・ 90 ページの[エイリアステーブルについて](#)
- ・ 81 ページの[テーブルキーについて](#)
- ・ 78 ページの[データファンデーションのテーブルについて](#)

## 6.7.3 テーブル名の大文字/小文字の設定

テーブル名をすべて大文字または小文字にする必要があるデータベースもあります。[大文字/小文字の設定] コマンドを使用して、テーブル名の大文字/小文字を変更します。

- 1 エディタでデータファンデーションを開くには、[ローカルプロジェクト] ビューでデータファンデーションを選択します。
- 2 [データファンデーション] ビューでテーブルヘッダを右クリックして、[大文字/小文字の設定] を選択します。次に、[大文字] または [小文字] を選択します。

### 注

複数のテーブルを選択するには、CTRL キーを押しながらテーブルヘッダをクリックします。



## 6.8 テーブルの値の表示および分析。

データファンデーションテーブルで値を参照できるコマンドがいくつかあります。コマンドにアクセスするには、データファンデーションビューでテーブルヘッダまたは列名を右クリックします。

- ・ [テーブルの値の表示] では、テーブルの値が一覧表示されます。特定の列値をフィルタし、表示された値をローカルファイルに保存できます。
- ・ [プロファイルデータ] では、選択した列の値を円グラフや棒グラフで分析できるダイアログボックスが開きます。
- ・ [列の値の表示] には、選択された列の値が表示されます。特定の列値をフィルタし、表示された値をローカルファイルに保存できます。

テーブルに定義された列フィルタは、値を表示する前に適用されます。

[値の表示] ビューに列フィルタを追加するには、[フィルタの追加] をクリックし、列を選択します。列にフィルタを追加すると、フィルタ値のセレクトが右側に表示されます。

選択された列の値を分析するには、[一意の値] タブをクリックします。

ローカルファイルにデータをエクスポートするには、[ファイルとして保存] をクリックします。

値を並べ替えるには、列名をクリックします。

データベースから返される行を制限するには、[最大行数] ボックスに数字を入力します。

クエリスクリプトを表示するには、値のテーブルの上にある矢印をクリックします。

### 注

値の表示コマンドは、デフォルトではエディタでタブを開き、値を表示します。コマンドによって専用のビューまたはダイアログボックスで開くよう基本設定を設定できます。基本設定を設定するには、このメインメニューから、[ウィンドウ] > [基本設定] > [インフォメーションデザインツール] > [データの表示] を選択します。

## 6.9 テーブルキーについて

データファンデーションのテーブルには 2 つのキーがあります。

キー	説明
プライマリ	値でテーブルの各行を識別するテーブルの列または列の組み合わせ。一次キーで、テーブルの各行が一意であることを保証します。各テーブルには、一次キーが 1 つだけあります。

キー	説明
外部	<p>他のテーブルのプライマリキーあるいは一意のキーに一致する値を持った1つあるいは複数の列の組合せ</p> <p>外部キーは、たとえば、〈Custome〉テーブルに存在しない顧客の〈Sales〉テーブルへの売上の追加を許可しないなどの、制約を実装します。各テーブルには複数の外部キーを設定できます。</p>

キーは、データファンデーションビューの列の隣のアイコンで示されます。

[キーの検出] コマンドは、データベーステーブルのキーを検出し、同じキーをデータファンデーションのテーブルに設定します。[テーブルの挿入] コマンドは、オプションとしてデータファンデーションにまずテーブルを挿入するときに自動的にキーを検出するオプションがあります。

列に[キーとして設定] コマンドを使用すると、データファンデーションテーブルのあらゆる列に一次または外部キーを定義できます。キーは、テーブルを編集するときに手動でも設定できます。

#### 注

次回[キーの検出] コマンドを使用するときに、データベーステーブルに定義されているキーが、テーブルに手動で設定されているキーを上書きします。

キーが検出されなかったときに、データファンデーションテーブルに手動で設定したキーを維持するよう基本設定を設定できます。基本設定を設定するには、[ウィンドウ] > [基本設定] > [インフォメーションデザインツール] > [データファンデーションエディタ] > [自動検出] を選択します。

## 6.10 表の行数について

### 行数の検出


データベーステーブル内の行数を検出してデータファンデーションに保存できます。行数は、テーブルキーのない場合のカーディナリティを検出するために使用されます。

行数を検出すると、選択したテーブルの行数が数えられ保存されます。

#### 注

行数の検出に列フィルタは適用されません。

テーブルの推定行列も設定できます。これは、データの縮小サンプルを使用しているときに、運用データのサイズにクエリを最適化するときに役立ちます。設定した行数は、そのテーブルの行数を検出したときに検出された行数によって置き換えられます。

[検出] メニューの[行数の検出 ] コマンドは、データファンデーション内のすべてのテーブルの現在の行カウントを表示します。この一覧から、行数を設定し、選択したテーブルの行数を検出します。

あるテーブルの行数を検出するには、データファンデーションビューでテーブルヘッダを右クリックし、[検出] > [行数] を選択します。選択したテーブルの行数が更新されます。

#### 注

複数のテーブルを選択するには、CTRL キーを押しながらテーブルヘッダをクリックします。

#### 行数のカウント

[行数のカウント] コマンドを、結合で連結された複数のテーブルに使用して、結果のクエリによって返される行数を表示します。列フィルタが適用されます。

クエリで返された行数を数えるには、データファンデーションビューで次のいずれかの方法でテーブルを選択します。

- ・ テーブルを右クリックして、[関連テーブルの選択] を選択します。
- ・ CTRL キーを押しながらテーブルヘッダをクリックします。

次に選択したテーブルを右クリックして [行数のカウント] を選択します。

#### 関連項目

- ・ 83 ページの [列フィルタについて](#)


## 6.11 列フィルタについて

列フィルタでは、テーブルがクエリで使用されているときに返される値を制限できます。

フィルタを挿入するには、データファンデーションビューの列名を右クリックし、[フィルタの挿入] を選択します。

フィルタを挿入すると、列の自己結合が [結合の編集] ダイアログボックスに提案されます。たとえば、<Customer> テーブルの <age> 列にフィルタを挿入すると、以下の自己結合が提案されます。

```
"Customer". "age" = "Customer". "age"
```

自己結合の 2 番目の部分 (等号の右にある式) を編集して列値をフィルタします。結合式の編集に関するヘルプを参照するには、[SQL アシスタント] アイコン  をクリックします。

列フィルタには、以下のルールが適用されます。

- ・ 1 つの列に 1 つのフィルタしか使用できません。
- ・ フィルタは計算列に挿入できません。
- ・ 式にはサブクエリを含められません。
- ・ @Prompt および @Variable の @ 関数を式で使用できます。
- ・ 標準テーブルにフィルタを挿入し、テーブルからエイリアスを作成すると、フィルタはエイリアステーブルに挿入されません。
- ・ エイリアステーブルにフィルタを挿入すると、フィルタは、元の標準テーブルに自動的に挿入されません。
- ・ フィルタを含むテーブルを結合すると、結果の派生テーブルにフィルタは含まれません。

## 6.12 計算列について

計算列は、同じテーブルの 1 つ以上の列に基づいた計算結果である、テーブルの新しい列です。

計算列を挿入するには、データファンデーションビューでテーブルヘッダを右クリックし、[計算列の挿入] を選択します。

テーブルに計算列を挿入するときに、列を定義する SQL SELECT 文を構築します。[計算列の挿入] ダイアログボックスには、SELECT 文を構築する枠にドラッグアンドドロップできる列と関数が表示されます。

計算列は以下のルールが適用されます。

- ・ 計算列は標準テーブルのみに挿入できます。
- ・ SELECT 文には、同じテーブルからの列しか含めません。
- ・ 使用できる関数が、[計算列の挿入] ダイアログボックスの関数枠に表示されます。
- ・ サブクエリは使用できません。
- ・ データファンデーションで複数のソースが有効化されており、計算列を定義するためにデータベース固有の SQL を使用する場合は、[データベース固有] チェックボックスにチェックを付けます。複数のソースが有効にされているデータファンデーションの SQL 式の詳細については、関連項目を参照してください。

テーブルに列が挿入されると、データファンデーションビューに特別なアイコンで表示されます。列名の上にカーソルを乗せると、計算列の SQL 式がツールヒントに表示されます。

テーブルに定義された列がある場合に定義を編集するには、データファンデーションビューのテーブルの列名を右クリックし、[計算列の編集] を選択します。

### 時間列

時間に関連するデータ型の列に基づいた日付部分（月、四半期、年など）を含む計算列を挿入できます。

時間列を挿入するには、時間に関連したデータ型の列を右クリックし、[時間列の挿入] を選択します。

### 注

時間に関連するデータ型の列には、カレンダーに似た特別なアイコンが付いています。

リストから、日付部分を選択します。計算列がテーブルに挿入されます。[計算列の編集] コマンドで新しい列を編集できます。

### 関連項目


- ・ 75 ページの [複数のソースが有効化されたデータファンデーションの SQL 式](#)

## 6.13 結合について

結合は、データファンデーションのテーブルをつなぐ条件です。結合は、2 つのテーブルがクエリされたときに返されるデータを制限します。


結合されるテーブルには、通常、親子関係があります。テーブルが結合されない場合、2 つのテーブルに実行されるクエリは、あらゆる行の組み合わせが含まれる結果セットを返します。このようなクエリ結果はデカルト積と呼ばれ、役に立つデータとなることはほとんどありません。

結合は、1 つのテーブルの列を 2 番目のテーブルの列につなげて定義されます。自己結合は、2 つのテーブルが同じ場合に起こります。自己結合は、列フィルタを定義するために使用されます。列フィルタの詳細については、関連項目を参照してください。

結合を挿入するには、データファンデーションビューの [挿入] メニューから [結合の挿入]  コマンドを選択します。

結合を挿入するときに、[結合の編集] ダイアログボックスで以下の結合プロパティを定義します。

結合プロパティ	説明
テーブル 1 の列	結合に使用する最初のテーブルの列です。
テーブル 2 の列	2 番目のテーブルの列、または結合先の列です。
結合演算子	<p>テーブル 1 とテーブル 2 の間の結合演算子のリストボックスでは、結合の列の値を比較する方法を選択できます。</p> <p>リストには等結合 (=) と、列の値の等価性を基準としない演算子 (&gt;, &gt;=, &lt;, &lt;=, !=) が含まれます。</p> <p>複雑な結合を作成することも選択できます。これはサブクエリを含む結合です。</p>
ショートカット結合	<p>ショートカット結合を作成するには、[ショートカット結合] チェックボックスにチェックを付けます。</p> <p>ショートカット結合は、2 つのテーブル間に代替パスを提供する結合です。ショートカット結合は、仲介テーブルを考慮せず、長い結合パスを短縮するので、クエリのパフォーマンスが向上します。</p>
外部結合	<p>外部結合を作成するには、[外部結合] チェックボックスにチェックを付けます。外部結合では、結合されたテーブルに一致する行がなくても、行を返すことができます。</p> <p>テーブル 1 の下の [外部結合] チェックボックスにチェックを付けると、左外部結合が作成されます。この結合では、テーブル 2 と一致しなくてもテーブル 1 のすべての行が返されます。</p> <p>テーブル 2 の下の [外部結合] チェックボックスにチェックを付けると、右外部結合が作成されます。この結合では、テーブル 1 と一致しなくてもテーブル 2 のすべての行が返されます。</p> <p>両方のテーブルの下の [外部結合] チェックボックスにチェックを付けると、完全外部結合が作成されます。この結合では、両方のテーブルからすべての行が返され、一致するものがない場合は行に null 値が設定されます。</p>

結合プロパティ	説明
式	選択した列と演算子を基に、結合を定義するために SQL 式が自動的に生成されます。結合のためのカスタム式を入力できます。結合式の編集に関するヘルプを参照するには、[SQL アシスタント] アイコン  をクリックします。
カーディナリティ	[カーディナリティ] リストボックスから結合のカーディナリティを選択します。また、[検出] ボタンをクリックすると、データベースで結合のために定義されているカーディナリティが自動的に検出されます。  カーディナリティの詳細については、関連項目を参照してください。

結合は自動的に検出できます。結合の検出の詳細については、関連項目を参照してください。

既存の結合を編集するには、データファンデーションビューの結合線を右クリックして、[結合の編集] を選択します。

#### 関連項目

- ・ 83 ページの [列フィルタについて](#)
- ・ 87 ページの [カーディナリティについて](#)
- ・ 86 ページの [結合の検出について](#)

### 6.13.1 結合の検出について


結合の検出では、データファンデーションテーブルを検索して適切な結合を提案します。次の方法が使用されます。

- ・ 列名を基にした結合検出。この方法では、異なるテーブルにある同じ列名が検索されます。また、2 つの列のデータ型が同じかどうかもチェックします。2 つのテーブル間で 1 つ以上の列が一致する場合は、各列の結合が提案されます。

#### 注

1 つのテーブル間の結合と、そのエイリアスは提案されません。

- ・ データベースキーを基にした結合検出この方法では、データベースに定義されている一次キーと外部キーの間の関係を検索します。
- ・ SAP NetWeaver BW 接続のあるデータファンデーションでは、結合検出は、接続で参照されているデータベーススキームの結合を基にしています。

データファンデーションで結合を検出するには、データファンデーションビューの [検出]  メニューから [結合の検出] を選択します。

まず、結合検出の方法を選択します。

複数のソースが有効化されているデータファンデーションでは、接続ごとに方法を選択します。この方法は、接続によって参照されているテーブル間の結合を検出するために使用されます。異なる接続からのテーブル間の結合も検出できます。この場合、使用される方法は列名です。

結合検出の方法を選択すると、結合が検出され、ダイアログボックスに提案されます。次に、データファンデーションに挿入する結合を選択できます。

テーブルを挿入するときには、結合を自動的に検出して挿入できます。自動結合検出のデフォルト、使用するデフォルトの検出方法をアプリケーションの基本設定ページで設定します。メインメニューから、[ウィンドウ] > [基本設定] > [インフォメーションデザインツール] > [データファンデーションエディタ] > [自動検出] を選択します。

#### 関連項目

- 84 ページの[結合について](#)

## 6.14 カーディナリティについて

カーディナリティは、1 つテーブルのいくつの行が他のテーブルの行と一致するかを示して、テーブルがどのように結合されているかをさらに詳しく説明します。カーディナリティは、データファンデーションでループを解決するためにエイリアスやコンテキストを検出するときに必要なになります。

テーブルのカーディナリティは、1 つテーブルのいくつの行が結合されたテーブルの行と一致するかを示す 1 組の数字で表されます。各テーブルの一致する行数には、なし (0)、1 (1)、または多数 (many) があります。

たとえば、<Customer> テーブルと <Reservations> テーブルが結合でリンクされていたとします。

- 顧客 1 人につき、1 つ以上の予約があることが考えられるため、<Customer> テーブルのカーディナリティは、1 対多 (1, n) になります。
- 予約 1 つにつき、顧客は 1 人しかいないため、<Reservations> テーブルのカーディナリティは、1 対 1 (1,1) になります。

結合のカーディナリティも、1 番目のテーブルの 1 つの行に一致する 2 番目のテーブルの最大行数と、2 つ目のテーブルの 1 つの行に一致する 1 番目のテーブルの最大行数を示す、1 組の数字で表されます。


たとえば、<Customer>-<Reservations> 結合のカーディナリティは、<Customer> テーブルの 1 つの行に一致する最大行数は n で、<Reservations> テーブルの 1 つの行に一致する最大行数は、1 であるため、n,1 になります。

結合のカーディナリティは自動的に検出されデータファンデーションに保存されます。検出メソッドによってまず一次キーおよび外部キーが検出されます。カーディナリティは、2 つのテーブルの列のキーステータスに従って以下のように設定されます。

1 番目のテーブルの列	2 番目のテーブルの列	カーディナリティ
プライマリキー	外部キー	1, n
外部キー	プライマリキー	n,1



キーが検出されない場合は、カーディナリティはテーブルの行カウントを使って設定されます。

カーディナリティを検出または設定するには、[検出] メニューの [カーディナリティの検出]  を選択します。[カーディナリティの検出] ダイアログボックスには、データファンデーションに保存されているすべての結合の最新のカーディナリティが表示されます。このリストから、選択した結合のカーディナリティを手動で設定したり検出できます。

1 つの結合のカーディナリティを検出するには、データファンデーションビューの結合線を右クリックし、[カーディナリティの検出] を選択します。選択された結合のカーディナリティが更新されます。

#### 注

複数の結合を選択するには、CTRL キー を押しながら結合線をクリックします。

カーディナリティは、[結合の編集] コマンドを使用して結合詳細を編集するときにも手動で設定できます。

結合を挿入するときはカーディナリティを自動的に検出して設定できます。アプリケーションの基本設定のページで、カーディナリティの自動検出のデフォルト情報を設定します。メインメニューから、[ウィンドウ] > [基本設定] > [インフォメーションデザインツール] > [データファンデーションエディタ] > [自動検出] を選択します。

#### 関連項目


- ・ 84 ページの[結合について](#)
- ・ 81 ページの[テーブルキーについて](#)

## 6.15 派生テーブルについて

派生テーブルは、計算と関数を使用して他のテーブルを組み合わせます。派生テーブルのビジネスレイヤに、標準テーブルと同じようにオブジェクトを作成できます。派生テーブルは以下の状況で使用します。

- ・ 他のテーブルからの列を使ってテーブルを作成する。列の定義に複雑な計算や関数が含まれることがあります。
- ・ 1 つ以上のテーブルを組み合わせる 1 つのテーブルを作成する (結合テーブルと呼ばれます)。結合テーブルの詳細については、関連項目を参照してください。
- ・ 異なるテーブルからの一部の列を含むテーブルを作成する。

元のテーブルのすべての列を含む派生テーブルを挿入するには、データファンデーションビューでテーブルヘッダを右クリックし、[挿入] > [派生テーブル] を選択します。

派生テーブルを挿入し列を指定するには、データファンデーションビューの [挿入] メニューから [派生テーブルの挿入]  を選択します。

テーブル定義を [派生テーブルの編集] ダイアログボックスに入力します。

データファンデーション内で一意の名前を派生テーブルに付けます。

複数のソースが有効にされているデータファンデーションでは、派生テーブルの定義にデータベース固有の関数を含めるには、データベース固有の構文を選択する必要があります。複数のソースが有効にされているデータファンデーションの SQL 構文の詳細については、関連項目を参照してください。



[テーブル] および [データベーステーブル] ボックスのテーブル一覧から列を [式] ボックスにドラッグして、派生テーブルに含めます。派生テーブルのための列は、クエリパネルと同じように動作する [SQLビルダ] を使用して選択することもできます。

式定義で使用できる関数は、[関数] ボックスに表示されます。関数は、[式] ボックスにドラッグできます。

派生テーブルを編集するには、派生テーブルのテーブルヘッダを右クリックし、[編集] を選択します。

#### 関連項目

- ・ 89 ページの [テーブルの結合について](#)
- ・ 75 ページの [複数のソースが有効化されたデータファンデーションの SQL 式](#)

## 6.16 テーブルの結合について

テーブルを結合すると、結合によって接続されている複数のテーブルからの結合された列で構成される派生テーブルをデータファンデーションに挿入できます。

テーブルを結合するには、データファンデーションビューで以下のいずれかの手順を実行して、結合するテーブルを選択します。

- ・ テーブルを右クリックして、[関連テーブルの選択] を選択します。
- ・ CTRL キーを押しながらテーブルヘッダをクリックします。

次に、選択したテーブルを右クリックして、[結合] を選択します。

データファンデーション内で一意なテーブル名を入力します。

結合されたテーブルが、派生テーブルとして挿入されます。新しいテーブルは、元のテーブルが結合されたテーブルに結合されます。元のテーブルは不要になり、削除することを選択できます。元のテーブルを維持することを選択した場合は、それらのテーブルをつなぐ結合は削除されますが、テーブルはテーブルファンデーションに残ります。

結合されたテーブルを編集するには、テーブルヘッダを右クリックして [派生テーブルの編集] を選択します。

#### 注

複数のソースが有効化されているデータファンデーションでは、結合の結果である派生テーブルは、SQL-92 標準構文を使って式を作ります。データベース固有の SQL を使用するには、派生テーブルを編集し、データベース固有の構文を明示的に選択する必要があります。

#### 関連項目


- ・ 88 ページの [派生テーブルについて](#)
- ・ 75 ページの [複数のソースが有効化されたデータファンデーションの SQL 式](#)


## 6.17 エイリアステーブルについて

エイリアステーブルは、データファンデーションの標準テーブルへの参照です。これは元のテーブルとまったく同じ（列フィルタを除く）複製ですが、違う名前が付いています。テーブル内のデータはベーステーブルとまったく同じですが、別の名前をつけることで、クエリのSQLでは2つの異なるテーブルとして扱われます。

エイリアステーブルを使って、データファンデーションの結合パスのループを分断することができます。ループの解決の詳細については、関連項目を参照してください。

エイリアステーブルは、テーブルの名前変更にも使用できます。データファンデーションとデータベースの間のリンクは、古いテーブル名を基にしています。テーブルに新しい名前を付けるためにエイリアスを作った場合、データベーステーブルへのリンクは維持されますが、エイリアステーブル名はデータファンデーションで使用されます。

データファンデーションのエイリアスを検出するには、[検出 

エイリアスは、データファンデーションエディタの [エイリアスとコンテキスト] ペインからも検出できます。[エイリアスの検出 

このコマンドは、結合パスを分析し、データファンデーションで検出されたループを分断するためのエイリアステーブルを提案します。次に、エイリアスを自動的に作成するよう選択できます。

エイリアステーブルを手動で挿入するには、元のテーブルを選択します。データファンデーションビューで、CTRL キーを押しながらテーブルヘッダをクリックします。右クリックし、[挿入] > [エイリアステーブル] を選択します。

エイリアステーブルの名前と説明を編集するには、データファンデーションのテーブルヘッダをクリックし、[編集] を選択します。

### 関連項目

- ・ 91 ページの [ループの解決](#)

## 6.18 コンテキストについて

コンテキストとは、有効なクエリパスを定義する結合の組み合わせです。コンテキストの最も一般的な使用法は、エイリアステーブルを作成してもループを解決できなかったときにデータファンデーションでループを解決することです。コンテキストのもう 1 つの使用方法は、複数のファクトテーブルがディメンションテーブルを共有する場合です。この場合、コンテキストはファクトテーブルごとに作成されます。


インフォメーションデザインツールでは、コンテキストは、結合パスを明示的に定義することでループ内の 1 つの結合パスを解決します。ユーザは、クエリ実行時に使用するコンテキストを入力するよう求められます。コンテキストは、不明瞭性に関連する結合のステータスを設定することで定義します。

コンテキストでは、結合には 3 つのステータスがあります。

- ・ 含まれる結合: スキームの不明瞭な部分では、コンテキストは、含まれる結合でパスを定義することでループを解決します。
- ・ 除外される結合: スキームの不明瞭な部分では、除外される結合は、コンテキストが取らないパスを定義します。
- ・ 中性結合は、コンテキストのクエリパスに常に含まれる、スキームの不明瞭でない部分です。明示的に含まれるまたは除外されない結合は中性です。

新しい結合またはテーブルがデータファンデーションに挿入されるときにはデフォルトで中性になります。新しいテーブルまたは結合が明示的に関連しない限り、コンテキストを更新する必要はありません。


データファンデーションでコンテキストを検出する前に、カーディナリティを設定し、エイリアスを検出する必要があります。ループを解決するため前提条件のタスクについては関連項目をご覧ください。

データファンデーション内のコンテキストを検出するには、データファンデーションエディタの [エイリアスとコンテキスト] ペインから、[コンテキストの検出] アイコン  を選択します。

#### 注

エイリアスを使ってループを解決できるというメッセージが表示されることがあります。ループの解決に関連する項目を参照してください。

このコマンドは、結合パスを分析し、[コンテキストの検出] ダイアログボックスにエイリアステーブルで解決できなかったループを解決するためのコンテキストが提案されます。データファンデーションビューで強調表示されているコンテキストを見るには、提案されたコンテキスト名をクリックします。提案されたコンテキストをデータファンデーションに挿入するには、その隣にあるチェックボックスにチェックを付けます。

コンテキストを手動で挿入するには、データファンデーションエディタの [コンテキスト] タブから、[コンテキストの挿入]  を選択します。コンテキストに結合やテーブルを含める、またはコンテキストから除外するには、データファンデーションビューでクリックするか、結合パスのリストで結合パスをクリックします。

#### 関連項目

- ・ 91 ページの [ループの解決](#)
- ・ 90 ページの [エイリアステーブルについて](#)

## 6.19 ループの解決


ループは、複数のパスがテーブルを結合するときに起こります。クエリから返された行は、それぞれのパスの結果が交差する部分になるため、返される行数が予想される数よりも少なくなります。


エイリアステーブルは、クエリで、パスごとに 1 回ずつ 同じテーブルを 2 回使用することで、ループを分断します。このため、クエリで返された行は、各パスの結果の結合になります。

サイクルは、ループで結合されるテーブルのすべてにカーディナリティ (1,n) があるときに起こるループです。この場合、[エイリアスの検出] コマンドは、どのテーブルのエイリアスを作成するか決定できません。

エイリアステーブルでループが解決できない場合は、コンテキストを使用します。コンテキストを使用して、どの結合パスを使用するかクエリに明示的に指示することで、不明瞭性を解決します。

以下の手順を実行して、データファンデーション内のループを解決します。

- 1 データファンデーションのすべての結合を検出して挿入します。
- 2 結合のカーディナリティを検出または設定します。
- 3 データファンデーションにサイクルがないことを確認します。これは、任意のループのカーディナリティを調べるか、ループ解決ステータスを最新表示することで実行できます (以下の手順参照)。
- 4 結合に (n,n) のカーディナリティがないことを確認します。エイリアスとコンテキストは検出できません。(n,n) 結合のカーディナリティを手動で設定します。
- 5 データファンデーションエディタの [エイリアスとコンテキスト] ペインから、エイリアスを検出します。
- 6 コンテキストを検出します。提案されたすべてのコンテキストを挿入することをお勧めします。
- 7 [ループ] ボックスで、すべてのループが解決されたか確認できます。[ループの表示] アイコン  をクリックします。

ループと考えられるものがボックスに一覧表示されます。ループが解決されているか確認するには、[ループ解決ステータスの最新表示] アイコン  をクリックします。

未解決のループの処理方法を提案するメッセージが表示されます。

ループ名の隣に緑色のチェックマークが付いているときはループが解決されています。

#### 関連項目

- ・ 86 ページの [結合の検出について](#)
- ・ 87 ページの [カーディナリティについて](#)
- ・ 90 ページの [エイリアステーブルについて](#)
- ・ 90 ページの [コンテキストについて](#)

## 6.20 入力列について

SAP NetWeaver BW 接続が参照するテーブルでは、時間依存のデータを処理するために、入力列がデータファンデーションの中のテーブルに自動的に挿入されます。

クエリ実行時に入力列を解決するために、キー日付と呼ばれるパラメータがデータファンデーションに挿入されます。デフォルトでは、クエリ実行時にキー日付パラメータは入力を要求されません。これは現在の日付に自動的に割り当てられます。プロンプトパラメータをデータファンデーションで編集できます。

入力列のプロンプトパラメータを編集するには、データファンデーションビューでテーブルを右クリックし、[入力列の編集] を選択します。

データファンデーションビューで任意の場所を右クリックして、[入力列の編集]を選択することもできます。このコマンドは、データファンデーションのすべての入力列を表示します。リストから、編集する入力列を選択できます。

#### 関連項目

- ・ 75 ページの [NetWeaver BW 接続を使ったデータファンデーション](#)

## 6.21 データファンデーション内のパラメータと値の一覧について

パラメータは、クエリの実行時に値を必要とするユニバース内の変数です。パラメータは、多くの場合、ユーザーに値の指定を促すために定義されており、この場合、プロンプトとして参照されます。

値の一覧は、ユニバース内のオブジェクトに関連付けられるデータ値の集合で、ユーザーがプロンプトの値を選択できるようにします。

パラメータおよび値の一覧をデータファンデーションに挿入できます。これらは、データファンデーション上に構築されたビジネスレイヤによって継承されますが、ビジネスレイヤでは変更できません。

パラメータや値の一覧を挿入するには、データファンデーションエディタの [パラメータおよび値の一覧] タブを使用します。ここでの手順は、ビジネスレイヤにパラメータや値の一覧を挿入するときと同じです。関連項目を参照してください。

#### 関連項目

- ・ 124 ページの [パラメータについて](#)
- ・ 126 ページの [値の一覧について](#)

## 6.22 データファンデーションのプロパティについて

データファンデーションのプロパティは、データファンデーション全体に適用されます。データファンデーションのプロパティは、データファンデーションエディタの [プロパティ] タブで編集します。

説明を入力したり編集したりできます。これは、データファンデーションを [新しいデータファンデーション] ウィザードで作成するときに入力できる説明です。

[概要] をクリックすると、データファンデーションにあるオブジェクト (テーブル、結合、コンテキストなど) の数を表示したダイアログボックスが表示されます。

SQL オプションは、データファンデーションにあるすべての SQL 式に適用されます。

- ・ [デカルト積を許可する] チェックボックスにチェックを付けると、データファンデーションテーブルや結合の設計によりデカルト積が生じる場合は、SQL が許可されます。

- ・ **注**  
デカルト積は、クエリに含まれる各テーブルの各行の可能な限りの組み合わせを含む結果セットです。ほとんどの場合、デカルト積の結果は間違っています。
- ・ データファンデーションにコンテキストが含まれる場合は、[各コンテキストに対して複数の SQL 文を生成] チェックボックスにチェックを付けます。これにより、ユーザは、クエリにコンテキストが関連する場合に、クエリパスを選択できます。
- ・ [SQL パラメータ] をクリックすると、データファンデーションに SQL 生成パラメータが設定されます。SQL 生成パラメータの定義とデフォルト値の詳細については、関連項目を参照してください。

#### 関連項目

- ・ 209 ページの[SQL 生成パラメータについて](#)
- ・ 90 ページの[コンテキストについて](#)

## 6.23 ローカル依存関係の表示

データファンデーション内のテーブルと列を変更する場合、[ローカル依存関係の表示] コマンドを使用します。このコマンドにより、テーブルまたは列に依存しているビジネスレイヤとそのオブジェクトが見つかります。

- 1 [データファンデーション] ビューでテーブルヘッダまたは列名を右クリックして、[ローカル依存関係の表示] を選択します。

#### 注


Ctrl キーを押しながら、複数のテーブルまたは列を選択することができます。

選択したテーブルおよび列に依存するビジネスレイヤが一覧表示されます。

- 2 依存オブジェクトを表示するビジネスレイヤを選択します。  
ダイアログボックスにデータファンデーションのテーブルおよび列、これらに依存するビジネスレイヤオブジェクトが一覧表示されます。
- 3 ビジネスオブジェクトを編集するには、[ビジネスレイヤとオブジェクト] ボックスでオブジェクト名をダブルクリックします。選択されたオブジェクトに絞込まれて、ビジネスレイヤが開きます。

## 6.24 データファンデーションの最新表示について

構造を最新表示すると、基本となるデータベーステーブルへの変更でデータファンデーションが更新されます。

構造の最新表示ウィザードを起動するには、データファンデーションエディタの [検出] メニューから [構造の最新表示 ] を選択します。


ウィザードは、以下の変更を検出し、それぞれ独自のダイアログボックスに表示します。それぞれについて、データファンデーションに、どの提案された変更を実行するか選択します。

- ・ データベースで削除されたデータファンデーションの表。ウィザードは、これらのテーブルと、関連する結合をデータファンデーションから削除することを提案します。
- ・ データベーステーブルで削除されたデータファンデーションテーブルの列。ウィザードは、これらの列とこれらの列を使用する結合を削除するために、データファンデーションの対応する各表を更新することを提案します。
- ・ データベースに追加された列。ウィザードは、これらの列を追加するために、データファンデーションの各対応するテーブルを更新することを提案します。
- ・ データベースで変更された列のデータ型。ウィザードは、データベースの列の型と異なるデータファンデーション内の各列のデータ型を更新することを提案します。

ウィザードは、概要ダイアログボックスに選択された変更を一覧表示し、最新表示を続行する前に確認することをユーザに求めます。

## 6.25 カスタムデータファンデーションビューについて

カスタムデータファンデーションビューは、データファンデーション [マスタ] ビューのサブセットです。多数のテーブルを含むデータファンデーションを編集しているときや、テーブルのサブセットを操作するときに、ビューを使用します。データファンデーションには、複数のカスタムビューを定義できます。

カスタムビューを挿入するには、[挿入 

ビューの名前を入力します。ビュー枠の下部に新しいタブが表示されます。最初は、ビューは空です。

ビューにテーブルを追加するには、テーブルヘッダを右クリックし、[ビューに追加] を選択します。複数のテーブルを選択して追加するには、CTRL キーを押しながらテーブルヘッダをクリックします。

ビューを挿入するもう 1 つの方法では、1 つ以上のテーブルを選択します。次に、右クリックして、[挿入] > [選択からの表示] を選択します。ビューが挿入され、選択したテーブルが含まれます。

テーブルの操作はすべてのビューからできます。すべての変更はデータファンデーションのすべてのビューに伝播されます。

ファミリへのテーブルの割り当てもすべてのビューに伝播されます。

### 関連項目


- ・ 70 ページの [データファンデーションエディタについて](#)

## 6.26 テーブルファミリについて

ファミリは、同じタイプのテーブルを視覚的にグループ化するために使用できる表示パラメータのセットです。たとえば、ファクトテーブルおよびディメンションテーブルに異なるファミリを定義できます。

表示パラメータには、背景色、テキストの色、フォントが含まれます。



ファミリを追加または編集するには、データファンデーションビューで[ファミリの編集] アイコン  をクリックします。

[ファミリの編集] ダイアログボックスでは、ファミリを追加したり、表示パラメータを編集できます。

ファミリ定義をエクスポートおよびインポートできます。エクスポートすると、インフォメーションデザインツールの異なるユーザ間で共有できるローカルフォルダにファイルが作られます。

ファミリを定義したら、ファミリにテーブルを割り当てて、テーブルがファミリとして表示されるようにします。


ファミリにテーブルを割り当てるには、データファンデーションビューでテーブルヘッダをクリックします。[ファミリ] リストボックスで、ファミリを選択します。

#### 注

ファミリの割り当てを解除するには、[ファミリなし] を選択します。

## 6.27 データファンデーションビューでのコメントの挿入




コメントとは、データファンデーションビュー内のどこにでも配置できるメモのことです。

コメントを挿入するには、データファンデーションビューで、[挿入] () メニューから [コメントの挿入] を選択します。

[コメントの編集] ダイアログボックスで、メモの表示パラメータを定義し、コメントテキストを入力します。

コメントは、現在のビューの左上角に挿入されます。ビュー内の任意の表示位置に、コメントをドラッグします。

## 6.28 データファンデーションでのテーブルおよび列の検索

- 1 検索パネルを開くには、[データファンデーション] ビューで [検索パネルの表示/非表示] アイコン  をクリックします。
- 2 デフォルトでは、テーブルを検索します。列を検索するには、[フィルタ] テキストボックスでアイコン  をクリックします。
- 3 検索を制限するには、いくつかの方法があります。
  - ・ [フィルタ] テキストボックスに検索するテキストを入力します。
  - ・ リストボックスで、接続、テーブルの種類、列の種類、ファミリ、およびコンテキストを選択します。検索条件に一致するテーブルが、[データファンデーション] ビュー内に強調表示されます。
- 4 一致するテーブルのみが表示されるように、ビューを変更するには、検索パネルの上部にある [検索オプション] アイコン  をクリックして [自動配置検索結果] を選択します。



**ヒント**

[選択時に中央へ] コマンドによりデータファンデーション表示上でズームを変更できるため、選択項目のすべてのテーブルは表示ウィンドウに表示されます。

- 5 [リセット] をクリックすると、検索条件がクリアされ、新しい検索を実行できます。


**関連項目**

- ・ 97 ページの[選択時のビューの中央揃え](#)

## 6.29 選択時のビューの中央揃え

[選択時に中央へ] コマンドにより[データファンデーションビュー] 画面上でズームを一時的に変更できるため、選択項目のすべてのテーブルは表示ウィンドウに表示されます。

- 1 データファンデーションエディタでテーブルの選択を作成します。  
たとえば、検索パネルを使用して特定のファミリーですべてのテーブルを選択します。
- 2 選択したテーブルのいずれかのテーブルヘッダを右クリックして、[選択時に中央へ] を選択します。

データファンデーション表示が拡大され、すべての選択されたテーブルが表示ウィンドウに表示されます。表示をリセットするには、検索パネルが開いている場合は閉じ、またはデータファンデーションビューの下ツールバーの [ズームのリセット] アイコン  をクリックします。

**ヒント**

[データファンデーション] パネルのツリービューでテーブル名または結合名を選択し、テーブルまたは結合の表示を画面の左に中央に揃えることもできます。

**関連項目**

- ・ 96 ページの[データファンデーションでのテーブルおよび列の検索](#)



## ビジネスレイヤの使用

### 7.1 ビジネスレイヤとは

ビジネスレイヤとは、データベース内の SQL または MDX 定義にマップされるメタデータオブジェクトのコレクションのことで、たとえば、列、ビュー、データベース関数、または事前に集計済みの計算などです。メタデータオブジェクトには、ディメンション、階層、メジャー、属性、および定義済みの条件が含まれます。各オブジェクトは、データを返すためにクエリで変更できるビジネスデータに対応します。ビジネスレイヤは、OLAP キューブまたはリレーショナルデータベースで作成されるデータファンデーション上で、直接作成できます。

ビジネスレイヤが完了すると、CMS にユニバースとして公開されます。ユニバースは .unx ファイルを公開し、これにはビジネスレイヤと OLAP キューブへの接続、またはビジネスレイヤと対応するデータファンデーションが含まれます。ユニバースは、リポジトリにおいて SAP BusinessObjects データ分析とレポート作成アプリケーションで使用できます。

ビジネスレイヤの基本的役割は、ユニバースとして公開される前にメタデータレイヤを定義して編成することです。別の方法でビジネスレイヤを理解するには、メタデータワークベンチと考えてください。メタデータワークベンチは、メタデータセットをデータ分析とレポート作成アプリケーション用にユニバースとして公開する前に、収集して変更するためにデザイナーが使用します。

#### 関連項目

- 99 ページの[ビジネスレイヤの作成](#)

### 7.2 ビジネスレイヤの作成

ビジネスレイヤの新規作成ウィザードを使用して、データファンデーションまたは OLAP キューブに基づくビジネスレイヤを作成します。

ビジネスレイヤを作成する前に、以下のことが必要です。

- ・ [ローカルプロジェクト] ビュー内のプロジェクトフォルダ
  - ・ データファンデーションでビジネスレイヤを基にしている場合、同じプロジェクトフォルダに保存されるデータファンデーション。
  - ・ OLAP キューブでビジネスレイヤを基にしている場合、同じプロジェクトフォルダに保存される OLAP 接続または接続ショートカット。
- 1 [ローカルプロジェクト] ビューでプロジェクトフォルダを選択します。

- 2 [ファイル] > [新規] > [ビジネスレイヤ] を選択します。
- 3 新規ビジネスレイヤウィザードの手順に従って、以下の情報を入力します。
  - ・ データファンデーションまたは OLAP 接続に基づくビジネスレイヤかどうか
  - ・ ビジネスレイヤの名前
  - ・ データソース
  - ・ OLAP キューブに基づくビジネスレイヤの場合、ビジネスレイヤに含まれるキューブからのオブジェクト。

ビジネスレイヤは、データファンデーションまたは OLAP キューブから選択されたオブジェクトに基づいて作成され、ビジネスレイヤエディタに表示されます。

#### 関連項目

- ・ 100 ページの[ビジネスレイヤの新規作成ウィザードの使い方](#)

## 7.2.1 ビジネスレイヤの新規作成ウィザードの使い方

ビジネスレイヤの新規作成ウィザードは、ユーザが以下の段階に沿ってビジネスレイヤを作成できるようにします。特定ページの詳しい説明は、該当する関連項目を参照してください。

#### 関連項目

- ・ 100 ページの[ビジネスレイヤのデータソースの種類の指定](#)
- ・ 101 ページの[ビジネスレイヤ名の指定](#)
- ・ 101 ページの[ビジネスレイヤのデータファンデーションの選択](#)
- ・ 102 ページの[ビジネスレイヤの OLAP 接続の選択](#)
- ・ 102 ページの[OLAP キューブからのビジネスレイヤのオブジェクトの選択](#)

### 7.2.1.1 ビジネスレイヤのデータソースの種類の指定

この節では、新しいビジネスレイヤウィザードのデータソースの種類の選択ページについて説明します。

リレーショナルまたは OLAP データソースのいずれから、ビジネスレイヤを作成するかを選択します。

データソースの種類	説明
リレーショナル	ビジネスレイヤはデータファンデーションに基づきます。現在のプロジェクトフォルダでは、あらゆるデータファンデーションを選択できます。
OLAP	ビジネスレイヤは OLAP キューブに基づきます。現在のプロジェクトフォルダでは、あらゆる OLAP 接続または接続ショートカットを選択できます。

- 1 一覧からデータソースの種類の内 1 つをクリックします。
- 2 [次へ] をクリックします。  
新しいビジネスレイヤに対して、ビジネスレイヤの名前を付けるページが表示されます。

#### 関連項目

- ・ 101 ページの[ビジネスレイヤ名の指定](#)

### 7.2.1.2 ビジネスレイヤ名の指定

この節では、新しいビジネスレイヤウィザードのリソース作成ページについて説明します。

ビジネスレイヤの名前と説明を入力します。これは、ビジネスレイヤから公開されるユニバースの名前です。

#### 関連項目

- ・ 101 ページの[ビジネスレイヤのデータファンデーションの選択](#)
- ・ 102 ページの[ビジネスレイヤの OLAP 接続の選択](#)

### 7.2.1.3 ビジネスレイヤのデータファンデーションの選択

この節では、新しいビジネスレイヤウィザードの [データファンデーション選択] ページについて説明します。

新しいビジネスレイヤのデータソースとして、データファンデーションを選択します。次の操作から 1 つを選択できます。

- ・ データファンデーション内のテーブルと列から、自動的にビジネスレイヤオブジェクトを作成する。
  - ・ 空のビジネスレイヤを作成する。作成後に、データファンデーションからオブジェクトを手動で追加する必要があります。
- 1 [データファンデーション] テキストフィールドの末尾のボタンをクリックします。  
利用可能なデータファンデーションの一覧が表示されます。
  - 2 一覧でデータファンデーションをクリックして、[OK] をクリックします。  
データファンデーション名が名前フィールドに表示されます。[クラスおよびオブジェクトの自動作成] チェックボックスはデフォルトで選択されています。
  - 3 次のいずれかを実行します。
    - ・ オブジェクトおよびクラスを含むビジネスレイヤを自動的に設定する場合、[完了] をクリックします。
    - ・ オブジェクトおよびクラスを含むビジネスレイヤを自動的に設定しない場合、チェックボックスをオフにして、[完了] をクリックします。ビジネスレイヤを手動で入力する必要があります。新しいビジネスレイヤが [編集] タブに表示されます。

## 関連項目

- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)

### 7.2.1.4 ビジネスレイヤの OLAP 接続の選択

この節では、新しいビジネスレイヤウィザードの [OLAP 接続選択] ページについて説明します。

新しいビジネスレイヤのデータソースとして、OLAP 接続とキューブを選択します。次のオプションが使用できます。

OLAP 接続オプション	説明
OLAP 接続	テキストフィールドの末尾にある [参照] ボタンをクリックして、プロジェクトで定義される OLAP 接続または接続ショートカットを選択します。
measure 集計関数の検出	選択されていない場合、依頼されたデータベース関数が適用されます。
技術名からの属性の作成	属性は、各ディメンションの技術名に対して作成されます。
検索	キューブの検索文字列を入力して、検索アイコンをクリックします。
接続キューブの一覧	接続に対して利用可能なキューブの一覧です。複数のキューブがある場合、キューブを参照して、ターゲットキューブを選択します。

## 関連項目

- ・ 102 ページの[OLAP キューブからのビジネスレイヤのオブジェクトの選択](#)

### 7.2.1.5 OLAP キューブからのビジネスレイヤのオブジェクトの選択

この節では、新しいビジネスレイヤウィザードのオブジェクトの選択ページについて説明します。

選択したキューブの下オブジェクトノードを展開し、新しいビジネスレイヤに含めるオブジェクトを選択します。選択し終わったら、[終了] をクリックします。

新しいビジネスレイヤが [ビジネスレイヤ] 枠に表示されます。

## 7.3 ビジネスレイヤエディタについて

ビジネスレイヤエディタを使用して、ビジネスレイヤオブジェクトとプロパティを作成および編集します。

ビジネスレイヤエディタは、左側の参照ペイン、右上の編集ペイン、右下のデータソースペインに分かれます。

参照ペインでは、ビジネスレイヤの異なる要素に作業できます。対応するタブをクリックしてペインにアクセスします。

- ・ ビジネスレイヤ
- ・ クエリ
- ・ パラメータおよび値の一覧
- ・ ナビゲーションパス

各参照ペインでの操作の詳細については、関連項目を参照してください。

[ビジネスレイヤ] は、デフォルトの参照ペインです。ビジネスレイヤ内のオブジェクトがツリービュー形式で表示されます。ビジネスレイヤのツリービューを表示して移動するために、次のオプションが使用できます。

- ・ ビジネスビューによるフィルタの適用
- ・ オブジェクトの検索
- ・ 表示オプションの変更: オブジェクトの表示/非表示、技術名の表示

編集ペインでは、参照ペインで選択されたオブジェクトまたは要素のプロパティを編集できます。

データソースペインには、データファンデーションまたは OLAP 接続情報が表示されます。

- ・ デフォルトでは、すべてのテーブルと結合が含まれるデータファンデーションマスタビューが表示されます。定義されている場合、別のデータファンデーションビューのタブが、データソースペインの最下部に表示されます。ビューを変更するには、タブをクリックします。
- ・ 接続内の OLAP メタデータがデータソースペインの左側に表示されます。ペインの右側でプロパティを表示するメタデータオブジェクトを選択します。

### 関連項目

- ・ 104 ページの[ビジネスレイヤのプロパティについて](#)
- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)
- ・ 119 ページの[ビジネスレイヤへのオブジェクトの挿入](#)
- ・ 119 ページの[データファンデーションからのオブジェクトの直接挿入](#)
- ・ 123 ページの[ビジネスレイヤのクエリオブジェクトについて](#)
- ・ 124 ページの[パラメータについて](#)
- ・ 126 ページの[値の一覧について](#)
- ・ 131 ページの[オブジェクトのナビゲーションパスについて](#)
- ・ 122 ページの[ビジネスレイヤビューについて](#)
- ・ 123 ページの[ビジネスレイヤビューによるフィルタの適用](#)
- ・ 120 ページの[ビジネスレイヤオブジェクトの検索](#)

- 121 ページの[ビジネスレイヤのツリービューの表示オプションの変更](#)

## 7.4 ビジネスレイヤのプロパティについて

以下のプロパティおよび制約は、ビジネスレイヤ全体に対して定義されます。制限は、公開済みユニバースに適用されます。

プロパティ		説明
名前		ビジネスレイヤの識別および、ビジネスレイヤが公開される ときのユニバース。
説明		ユニバースの目的およびコンテンツの説明。この説明は、 公開されたユニバースを使用するクエリツールおよびレ ポートツールに表示できます。
クエリ制限	結果セットサイズ制限	クエリから返される行数を指定します。返される行数は制 限されるようになりますが、RDBMS ではクエリに含まれる すべての行が処理されます。RDBMS での処理結果が返 される行数のみが制限されます。
	最大実行時間	クエリの実行時に渡される実行時間を分で指定しますが、 データベースでの処理は停止しません。
	警告表示までの時間	ユーザは、推定実行時間が指定した時間（分単位）を超 過すると、メッセージを受信します。
クエリ（データ ファンデーションに 基づくビジネスレイ ヤに適用）	サブクエリの使用	サブクエリは、クエリ内で許可されています。
	ユニオン、インターセクト、マ イナス演算子の使用	ユニバースユーザは、データセット演算（ユニオン、イン ターセクト、マイナス）を使ってクエリを組み合わせ、結果 セットを受信することができます。
	クエリパネルでの複合演算 子の使用	クエリパネルでフィルタを定義するときに使用可能なオペ ランドの一覧に複合オペランドを使用できます。
要約		ビジネスレイヤで定義された各種のオブジェクトの数を表 示します。また、基になるデータソースで定義されたデー タファンデーションオブジェクトの数も表示します。
詳細	データソース	ビジネスレイヤのデータソースを、データファンデーション または OLAP 接続のいずれかから指定します。  OLAP データソースには、別のプロパティが定義されます。 詳細については、関連項目を参照してください。





プロパティ		説明
パラメータ (データファン デーションに基 づくビジネスレ イヤに適用)		SQL 生成パラメータのカスタム値を指定します。これは、データファンデーションのプロパティにあるデフォルト値またはカスタマイズされた値を上書きします。詳細については、関連項目を参照してください。
コメント		ビジネスレイヤについてのコメント

#### 関連項目

- ・ 106 ページの[ビジネスレイヤのプロパティの編集](#)」
- ・ 105 ページの[OLAP データソースプロパティ](#)」
- ・ 209 ページの[SQL 生成パラメータについて](#)」

### 7.4.1 OLAP データソースプロパティ

次のプロパティがビジネスレイヤの OLAP データソースに適用されます。

プロパティ	説明
OLAP 接続	<p>OLAP データソースへのアクセスを提供する接続または接続ショートカット。</p> <p>接続を変更するには、フィールドの最後にある参照アイコン (  ) をクリックして、使用可能な接続の一覧を開きます。</p>
キューブ	<p>現在の接続で選択されているキューブ。接続の定義時にキューブが指定されていなかった場合のみ、異なるキューブを選択できます。</p> <p>キューブを変更するには、フィールドの最後にある参照アイコン (  ) をクリックして、使用可能なキューブの一覧を開きます。</p>
END_MDX 値	<p>END_MDX パラメータの値。</p> <p>END_MDX パラメータは、データファンデーションに基づくユニバースで使用可能な END_SQL パラメータと同じです。END_MDX の値は、各 MDX 文の最後に追加されます。</p> <p>たとえば、END_MDX パラメータを使用して、クエリを実行中の人をトレースすることによって、データベースサーバのアクティビティを追跡できます。このソリューションには、各 MDX クエリの最後に、ユーザとユニバースに関する情報がコメントに追加されています。以下はその例です。</p> <pre>//User: @Variable('BOUSER') Universe: @Variable('UNVNAME')</pre>
メジャー階層名	このプロパティは、現在使用されていません。

## 7.4.2 ビジネスレイヤのプロパティの編集

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤをダブルクリックします。
- 2 ビジネスレイヤの最上位レベルが、[ビジネスレイヤ] ペイン内のツリービューで選択されていることを確認します。
- 3 右側の編集ペインで、ビジネスレイヤのプロパティを編集します。
  - ・ ビジネスレイヤの概要を確認するには、[概要] をクリックします。
  - ・ ビジネスレイヤのデータソースを編集するには、[詳細設定] をクリックします。OLAP データソースの高度なプロパティの詳細については、関連項目を参照してください。
  - ・ SQL 生成パラメータ値を編集するには、[パラメータ] をクリックします。
  - ・ ビジネスレイヤのコメントを入力または編集するには、[コメント] をクリックします。


- 4 ビジネスレイヤを保存し変更を適用します。

#### 関連項目

- ・ 104 ページの[ビジネスレイヤのプロパティについて](#)
- ・ 105 ページの[OLAP データソースプロパティ](#)

### 7.4.3 ビジネスレイヤのデータソースの変更

ビジネスレイヤのデータソースを変更するには、新しいデータソース(データファンデーション、OLAP 接続または接続ショートカット) がビジネスレイヤとして同じローカルプロジェクトフォルダに保存されている必要があります。

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト]ビューでビジネスレイヤ名をダブルクリックします。
- 2 ビジネスレイヤの最上位レベルが、[ビジネスレイヤ] ペイン内のツリービューで選択されていることを確認します。  
右側の編集ペインに、ビジネスレイヤプロパティが表示されます。
- 3 [詳細設定] ボタンをクリックします。
- 4 ビジネスレイヤのデータソースの種類によって、次のうちいずれかを実行します。
  - ・ データソースがデータファンデーションの場合、一覧から新しいデータファンデーションを選択して[OK]をクリックします。
  - ・ データソースが OLAP の場合、[OLAP 接続] テキストボックスの最後にある [参照] アイコン  をクリックします。新しい OLAP 接続を選択し、[OK] をクリックします。

#### 注

高度な OLAP プロパティの詳細については、関連項目を参照してください。

#### 関連項目

- ・ 105 ページの[OLAP データソースプロパティ](#)

## 7.5 ビジネスレイヤオブジェクトの使用

### 7.5.1 ビジネスレイヤオブジェクトについて

ビジネスオブジェクトペインには、ビジネスレイヤを構築するために使用するメタデータオブジェクトが含まれています。ビジネスレイヤのデータソースの種類によって、以下の種類のオブジェクトをビジネスレイヤで作成および編集できます。

- ・ ディメンション
- ・ メジャー
- ・ 階層 (OLAP のみ)
- ・ 分析ディメンション (OLAP のみ)
- ・ 属性
- ・ フィルタ
- ・ 名前付きメンバーの集合 (OLAP のみ)
- ・ 計算メンバー (OLAP のみ)
- ・ フォルダ

ビジネスレイヤ内の各オブジェクトには、いつでも定義および変更可能なプロパティがあります。ビジネスレイヤでオブジェクトに設定したプロパティは、公開されるユニバースに適用されます。

次のプロパティは、ビジネスレイヤ内のすべてのオブジェクトで共通です。

プロパティ	定義
名前	オブジェクトの名前。名前は、ターゲットユーザプロファイルのクエリおよびデータ分析カルチャーに対応している必要があります。オブジェクト名の入力時には、ユーザプロファイルに分かりやすいビジネス用語を使用します。
説明	ユニバースについて説明するコメントです。
有効/非表示/ 使用停止	<ul style="list-style-type: none"> <li>・ 有効: オブジェクトはクエリパネルに表示されます。</li> <li>・ 非表示: オブジェクトは有効ですが、クエリパネルでは使用できません (非表示オブジェクトとして別のオブジェクトで使用)。</li> <li>・ 使用停止: オブジェクトは非表示で使用できません。たとえば、ターゲットデータベースフィールドはすでに存在しないが、今後使用するときのためにオブジェクトを保持しておきたい場合。</li> </ul>

特定のビジネスレイヤオブジェクトとそのプロパティの詳細については、関連項目を参照してください。

#### 関連項目

- ・ 109 ページの[ディメンションおよびメジャーについて](#)
- ・ 110 ページの[階層について](#)
- ・ 111 ページの[分析ディメンションについて](#)
- ・ 111 ページの[属性について](#)
- ・ 112 ページの[フィルタについて](#)
- ・ 113 ページの[名前付きセットについて](#)
- ・ 114 ページの[計算メンバーについて](#)
- ・ 115 ページの[フォルダについて](#)

### 7.5.1.1 ディメンションおよびメジャーについて

ディメンションとメジャーは、ビジネスレイヤのメタデータの構成要素です。

ディメンションは、データベースで1つ以上のテーブル列または関数にマップされるオブジェクトで、クエリ内の分析の軸を表します。たとえば、製品、地域、時間、従業員が共通のディメンションです。各ディメンションは、ビジネス環境におけるアクティビティの側面を分類します。

メジャーは、計算を表すオブジェクトで、データベース内の統計および分析データにマップされる集計関数です。

ビジネスレイヤでは、ディメンションは分析の文脈情報（軸）、メジャーは事実情報（データ）を表します。

分析ディメンションでは、ビジネス上の疑問についての多次元分析が可能です。詳細については、関連項目を参照してください。

ディメンションとメジャーには次のプロパティを設定できます。

プロパティ	説明
データ型	オブジェクトのデータ型。事前定義済みの一覧からタイプを選択できます。
集計関数	メジャーは、メジャーオブジェクトによって返された数値情報の集計方法を定義します。詳細については、関連項目を参照してください。
キー	データファンデーションに基づくディメンションでは、1次キーおよび外部キーとして使用されるデータベース列を定義します。詳細については、関連項目を参照してください。
[SQL 定義] または [MDX 定義]	オブジェクトを定義するクエリ式。詳細については、関連項目を参照してください。
詳細	オブジェクト、表示の基本設定に適用されるクエリ制限、制約、値の一覧の設定を含むプロパティ。詳細については、関連項目を参照してください。
ソース情報	関連項目を参照してください。
カスタムプロパティ	関連項目を参照してください。

編集ペインにある [スクリプトの表示] および [値の表示] ボタンで、ディメンションのクエリスクリプトおよびデータを確認できます。

## 関連項目

- ・ 111 ページの[分析ディメンションについて](#)
- ・ 118 ページの[集計関数について](#)
- ・ 116 ページの[キービジネスレイヤオブジェクトのプロパティ](#)
- ・ 115 ページの[SQL 定義ビジネスレイヤオブジェクトのプロパティ](#)
- ・ 116 ページの[MDX 定義ビジネスレイヤオブジェクトのプロパティ](#)
- ・ 116 ページの[ビジネスレイヤオブジェクトのプロパティの詳細設定](#)
- ・ 118 ページの[ソースビジネスレイヤオブジェクトのプロパティ](#)
- ・ 118 ページの[ビジネスレイヤオブジェクトのカスタムプロパティ](#)
- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)

## 7.5.1.2 階層について

階層は、OLAP キューブに基づくビジネスレイヤでのみ使用できます。

階層は、OLAP キューブの階層のビジネスレイヤでの表現です。キューブでの階層がレベルに基づく場合、ビジネスレイヤのレベルオブジェクトは、レベルを表します。キューブでの階層が値に基づく場合、レベルはビジネスレイヤでは表示されません。レベルは、メンバーのプレビュー時と、メンバーセレクトアで表示できます。

編集パネルの [プレビュー] ボタンでは、階層内のメンバーの値を表示できます。

階層には、次を含むことができます。

- ・ レベル
- ・ 属性
- ・ 名前付きセット
- ・ 計算メンバー

階層には次のプロパティを設定できます。

プロパティ	説明
データ型	階層レベルオブジェクトに適用されます。オブジェクトのデータ型。
MDX 定義	階層を定義する MDX 式。詳細については、関連項目を参照してください。
詳細	オブジェクト、表示の基本設定に適用されるクエリ制限、制約、値の一覧の設定を含むプロパティ。詳細については、関連項目を参照してください。
ソース情報	関連項目を参照してください。
カスタムプロパティ	関連項目を参照してください。

## 関連項目

- ・ 116 ページの[MDX 定義ビジネスレイヤオブジェクトのプロパティ](#)」
- ・ 116 ページの[ビジネスレイヤオブジェクトのプロパティの詳細設定](#)」
- ・ 118 ページの[ソースビジネスレイヤオブジェクトのプロパティ](#)」
- ・ 118 ページの[ビジネスレイヤオブジェクトのカスタムプロパティ](#)」
- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)」

### 7.5.1.3 分析ディメンションについて

分析ディメンションでは、同じ分析の軸を共有するディメンションを論理的にグループ化できます。分析ディメンションは、一般的に、階層的な分析を表す場合に使用します。

**注**

分析ディメンションは、OLAP キューブに基づくビジネスレイヤでのみ使用できます。

分析ディメンションには次のプロパティを設定できます。

プロパティ	説明
型	このプロパティは、現在使用されていません。
デフォルト階層	クエリパネルで分析ディメンション全体が結果オブジェクトとして選択されている場合に、デフォルトとして使用される階層。
キー属性	このプロパティは、現在使用されていません。
カスタムプロパティ	関連項目を参照してください。

## 関連項目

- ・ 118 ページの[ビジネスレイヤオブジェクトのカスタムプロパティ](#)」
- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)」

### 7.5.1.4 属性について

属性は、親に関する追加の説明情報を提供する、親オブジェクトに添付されたオブジェクトです。属性は、次のディメンション、階層、およびレベルで定義することができます。

属性に固有の以下のプロパティを設定できます。

プロパティ	説明
ディメンション	属性の親ディメンションを選択できます。
[SQL 定義] または [MDX 定義]	オブジェクトを定義するクエリ式。詳細については、関連項目を参照してください。
詳細	オブジェクト、表示の基本設定に適用されるクエリ制限、制約、値の一覧の設定を含むプロパティ。詳細については、関連項目を参照してください。
ソース情報	関連項目を参照してください。
カスタムプロパティ	関連項目を参照してください。

#### 関連項目

- ・ 115 ページの[SQL 定義ビジネスレイヤオブジェクトのプロパティ](#)」
- ・ 116 ページの[MDX 定義ビジネスレイヤオブジェクトのプロパティ](#)」
- ・ 116 ページの[ビジネスレイヤオブジェクトのプロパティの詳細設定](#)」
- ・ 118 ページの[ソースビジネスレイヤオブジェクトのプロパティ](#)」
- ・ 118 ページの[ビジネスレイヤオブジェクトのカスタムプロパティ](#)」
- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)」

### 7.5.1.5 フィルタについて

フィルタとは、クエリで返されるデータを制限する条件オブジェクトです。フィルタは、クエリに適用されるように、クエリパネルの [クエリフィルタ] ペインに挿入することができます。

ネイティブフィルタは、データファンデーションテーブルで SQL の WHERE 句によって定義されます。ネイティブフィルタは、データファンデーションに基づくビジネスレイヤに適用します。

ビジネスフィルタは、ビジネスレイヤ内のディメンションおよびメジャーで、条件を作成および結合することによって定義されます。OLAP 接続に基づくビジネスレイヤの場合、ビジネスフィルタはメンバーの名前付きセットを使用して定義されます。

フィルタには次のプロパティを設定できます。

プロパティ	説明
フィルタタイプ	[ネイティブ] (データファンデーションに基づくビジネスレイヤのみ)、または [ビジネス] のいずれかです。



プロパティ	説明
SQL 定義	ネイティブフィルタの条件を定義する SQL 式。詳細については、関連項目を参照してください。
フィルタの定義	ビジネスフィルタの条件を定義するビジネスオブジェクトの選択および値制限。
プロパティ	<p>[クエリで必須としてフィルタを使用する] オプションが選択されている場合、選択された範囲 ([ユニバースに適用] または [フォルダに適用]) に従って、ユニバースまたはフォルダ内のオブジェクトを使用する各クエリにフィルを適用します。</p> <p>LOV クエリにもフィルタを適用するには、[値の一覧に適用] オプションを選択します。</p> <p>[クエリで必須としてフィルタを使用する] が選択解除されている場合、フィルタはクエリに明示的に追加されている場合のみ適用します。</p>
カスタムプロパティ	関連項目を参照してください。

#### 関連項目

- ・ 115 ページの[SQL 定義ビジネスレイヤオブジェクトのプロパティ](#)
- ・ 118 ページの[ビジネスレイヤオブジェクトのカスタムプロパティ](#)

### 7.5.1.6 名前付きセットについて

名前付きセットは、ビジネスレイヤ内の階層のメンバーの集合です。ネイティブの名前付きセットは、MDX 式を使用して定義されます。ビジネスの名前付きセットは、メンバーを選択することによって定義されます。

名前付きセットには次のプロパティを設定します。

プロパティ	説明
階層	名前付きセットの階層を選択できます。
集合の種類	[ネイティブ] または [ビジネス] のいずれかです。
MDX 定義	ネイティブの名前付きセットを定義する MDX 式。詳細については、関連項目を参照してください。

プロパティ	説明
定義	メンバーセクタを使用してビジネスセットを選択できます。メンバーを選択するには、[アイテムの編集]をクリックします。以前選択したメンバーを削除するには、一覧からそのメンバーを選択して[削除]をクリックします。

#### 関連項目

- ・ 141 ページの[メンバーセクタについて](#)
- ・ 116 ページの[MDX 定義ビジネスレイヤオブジェクトのプロパティ](#)
- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)

### 7.5.1.7 計算メンバーについて

計算メンバーは、OLAP キューブ、数理的演算子、数字、関数からのデータを含む明示的に定義された MDX 式を使って計算された階層のメンバーです。

計算メンバーは、クエリの作成時に使用されているメンバーセクタで使用できます。

計算メンバーには次のプロパティを設定できます。

プロパティ	説明
階層	計算メンバーを挿入する階層。
親メンバー	計算メンバーを挿入する階層内の場所。新しいメンバーの親を定義します。
MDX 定義	<p>計算メンバーを定義する MDX 式。詳細については、関連項目を参照してください。</p> <p>MDX クエリに含まれる次の MDX 計算と書式プロパティの値を入力できます。</p> <ul style="list-style-type: none"> <li>・ 解決順序</li> <li>・ 書式文字列</li> <li>・ 連結範囲分離</li> <li>・ 言語</li> </ul>

#### 関連項目

- ・ 141 ページの[メンバーセクタについて](#)
- ・ 116 ページの[MDX 定義ビジネスレイヤオブジェクトのプロパティ](#)

- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)

### 7.5.1.8 フォルダについて

フォルダは、関連オブジェクトのグループを保持するコンテナです。作成したフォルダには、ビジネスレイヤ内で共通の目的を持ったオブジェクトを格納および編成します。フォルダにはクエリ内でのロールはなく、オブジェクトを編成するためだけに使用されます。

フォルダには次のプロパティを設定できます。

フォルダプロパティ	説明
コンテンツ	フォルダ内のオブジェクトの一覧。クエリでのオブジェクトの使用目的を説明するプロパティを定義できます ([結果]、[フィルタ]、[並べ替え])。上下の矢印キーを使用して、フォルダ内の順序またはオブジェクトを一覧の右側へ変更できます。
カスタムプロパティ	関連項目を参照してください。

#### 関連項目

- ・ 118 ページの[ビジネスレイヤオブジェクトのカスタムプロパティ](#)
- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)

### 7.5.1.9 SQL 定義ビジネスレイヤオブジェクトのプロパティ

ビジネスレイヤオブジェクトのプロパティにある [SQL 定義] タブで、選択されたオブジェクトの SQL 文を定義できます。次のプロパティを入力できます。

プロパティ	説明
SELECT:	テキストボックスに直接 SELECT 句を入力するか、[SQL アシスタント] ボタンをクリックし、SQL エディタを使用して SELECT 句を作成します。
WHERE 句:	テキストボックスに直接 WHERE 句を入力するか、[SQL アシスタント] ボタンをクリックし、SQL エディタを使用して WHERE 句を作成します。

プロパティ	説明
追加テーブル	実行時にクエリに含めるオブジェクトに関連付けられているテーブルを選択できます。テキストフィールドの最後にあるボタンをクリックして、関連テーブルの一覧を開きます。追加テーブルを選択または消去します。

#### 7.5.1.10 キービジネスレイヤオブジェクトのプロパティ

ビジネスレイヤオブジェクトのプロパティにある [キー] タブで、データベース列が 1 次キーおよび外部キーであるディメンションオブジェクトを指定します。これにより、クエリがキー列でインデックスを使用できるようになります。キーを定義すると、クエリで生成される SQL を最適化することにより、データの取得がスピードアップします。たとえば、スタースキーマデータベース内で、ディメンションテーブルの値にフィルタを適用するようなクエリを作成すると、ディメンションテーブルの外部キーを使ってファクトテーブルに直接フィルタを適用できます。これにより、ディメンションテーブルの不要な結合を排除できます。

##### 注

キーの定義は、データファウンデーションで作成されたディメンションでのみ使用できます。

##### 関連項目

- 121 ページの [オブジェクトのキーの定義](#)

#### 7.5.1.11 MDX 定義ビジネスレイヤオブジェクトのプロパティ

多次元式 (MDX) は OLAP データベースへのアクセスに使用されるクエリ言語です。OALP キューブ上で作成されたディメンションおよびメジャーに対して SELECT 句および WHERE 句を定義するには、MDX を使用します。

ビジネスレイヤオブジェクトのプロパティにある [MDX 定義] タブで、選択されたオブジェクトの MDX 式を定義できます。テキストボックスに直接式を入力するか、[MDX アシスタント] ボタンをクリックし、MDX エディタを使用して式を作成します。

#### 7.5.1.12 ビジネスレイヤオブジェクトのプロパティの詳細設定

ビジネスレイヤオブジェクトのプロパティにある [詳細設定] タブでは、選択されたオブジェクトのプロパティの詳細設定を定義できます。次のプロパティを入力できます。

プロパティ	説明
アクセスレベル	<p>オブジェクトのセキュリティアクセスレベルを定義します。適切なセキュリティレベルを持つエンドユーザのオブジェクトの使用を制限するセキュリティレベルを選択できます。設定できるセキュリティアクセスレベルは次の通りです。</p> <ul style="list-style-type: none"> <li>・ パブリック</li> <li>・ プライベート</li> <li>・ コントロール</li> <li>・ リストリクト</li> <li>・ コンフィデンシャル</li> </ul> <p>パブリックレベルでは、すべてのユーザがそのオブジェクトを使用できます。[リストリクト]を割り当てると、リストリクトまたはそれ以上の権限をもつユーザだけがオブジェクトを表示し、使用できます。</p>
[オブジェクトの使用可能場所] [結果]	このオプションを選択すると、オブジェクトはクエリで使用できるようになります。
[オブジェクトの使用可能場所] [条件]	このオプションを選択すると、オブジェクトは条件で設定できるようになります。
[オブジェクトの使用可能場所] [並べ替え]	このオプションを選択すると、取得した値を並べ替えられるようになります。
データベース形式	<p>このプロパティは日付型のオブジェクトでのみ使用できます。</p> <p>デフォルトで、オブジェクトの日付書式は MS Windows のコントロールパネルの[地域のオプション]ダイアログボックスで定義されています。ターゲットデータベース形式の日付に適応するように、書式を変更することもできます。たとえば、日付書式をアメリカ形式やヨーロッパ形式にすることもできます。</p>
値の一覧	値の一案 (LOV) をオブジェクトに関連付けることができます。LOV は、クエリパネルでオブジェクトのフィルタを定義するときに適用します。
表示	<p>クエリのオブジェクトによって返されたデータの表示オプションを設定します。[表示書式の編集]をクリックして、事前定義済みの形式を選択するか、カスタム形式を定義します。</p> <p>また、オブジェクトによって返されたデータを [HTML] または [ハイパーリンク] として表示するように選択することもできます。</p>

### 7.5.1.13 ソースビジネスレイヤオブジェクトのプロパティ

ビジネスレイヤオブジェクトのプロパティ内の [ソース情報] タブには、Data Integrator で使用されるオブジェクトにのみ適用される説明フィールドが含まれます。

プロパティ	説明
技術情報	たとえば、オブジェクトに関連する列のオリジナルのデータベース名など、列に関する情報です。
Mapping	たとえば、売上げ = 複数ソースから計算された列などの、列の指定方法 (Data Integrator 使用による) を説明する初期式の情報です。
リネージュ	データベース内の列を計算するのに使用される式のソース列です。

### 7.5.1.14 ビジネスレイヤオブジェクトのカスタムプロパティ

ビジネスレイヤオブジェクトのプロパティにある [カスタムプロパティ] タブでは、選択されたオブジェクトのカスタムプロパティを定義できます。

オブジェクトのカスタムプロパティを追加または編集するには、以下の手順に従います。

- 1 エディタでビジネスレイヤを開き、[ビジネスレイヤ] ペインでオブジェクトを選択します。
- 2 編集ペインで、[カスタムプロパティ] タブを選択します。
- 3 カスタムプロパティを追加するには、[追加] をクリックします。
- 4 一覧内の列をクリックして、プロパティオブジェクト名と番号を編集します。
- 5 プロパティを削除するには、一覧からそのプロパティを選択して [削除] をクリックします。

### 7.5.1.15 集計関数について

集計関数は、メジャーオブジェクトによって返された数値情報の集計方法を定義します。よく使用される集計関数は、Sum、Count、Average、Minimum、Maximum、First、Last です。

Delegated 関数が選択されている場合は、メジャーの集計関数がデータベースから自動的に推定されます。


## 関連項目

- ・ 109 ページの[ディメンションおよびメジャーについて](#)

## 7.5.2 ビジネスレイヤへのオブジェクトの挿入

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤを選択します。
- 2 新しいオブジェクトを挿入する [ビジネスレイヤ] ペインのビジネスレイヤツリーで、オブジェクトを選択します。

最上位にオブジェクトを挿入するには、ツリーの最上位ノード (ビジネスレイヤ名) を選択します。

- 3 [ビジネスレイヤ] ペインの上部にある [オブジェクトの挿入] アイコン  をクリックし、挿入するオブジェクトの種類を選択します。

**注**

挿入可能なオブジェクトの一覧には、選択した親オブジェクトの下に挿入できるオブジェクトのみが含まれます。たとえば、ディメンションを選択すると、挿入メニューには、ディメンションに挿入可能な子オブジェクトとして属性のみが一覧表示されます。

- 4 新しいオブジェクトをクリックします。  
新しいオブジェクトのプロパティは、編集ペインにあります。新しいオブジェクトのプロパティを編集できます。オブジェクトのプロパティの詳細については、関連項目を参照してください。

## 関連項目

- ・ 107 ページの[ビジネスレイヤオブジェクトについて](#)

## 7.5.3 データファンデーションからのオブジェクトの直接挿入

データファンデーションに基づくビジネスレイヤの場合、オブジェクトをデータファンデーションからビジネスレイヤにドラッグアンドドロップできます。

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤ名をダブルクリックします。  
ビジネスレイヤがベースにしているデータファンデーションは、[編集] タブの右下にある [データソース] ペインに表示されます。
- 2 データファンデーションのビューで、挿入するオブジェクトを選択します。
  - ・ テーブルを選択するには、テーブルヘッダをクリックします。
  - ・ 複数のテーブルを選択するには、CTRL キーを押しながらテーブルヘッダをクリックします。
  - ・ 列を選択するには、テーブルで列名をクリックします。

- ・ 複数の列を選択するには、CTRL キーを押しながら列名をクリックします。
- 3 選択した列を [ビジネスレイヤ] ペインにドラッグし、ビジネスレイヤ内の目的のフォルダにそれらをドロップします。





**注**

テーブルをドロップすると、フォルダがビジネスレイヤに自動的に挿入され、すべての列のオブジェクトが含まれます。

## 7.5.4 ビジネスレイヤオブジェクトの編集

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤ名をダブルクリックします。  
ビジネスレイヤオブジェクトが、[ビジネスレイヤ] ペインに表示され、プロパティが右側の編集ペインに開きます。
- 2 [ビジネスレイヤ] ペインでオブジェクトを選択して、そのプロパティを編集します。
- 3 ユニバースを保存し変更を適用します。


## 7.5.5 ビジネスレイヤオブジェクトの検索

- 1 エディタでビジネスレイヤを開きます。  
[ビジネスレイヤ] 参照ペインには、ビジネスレイヤ内のオブジェクトがツリービュー形式で表示されます。
- 2 [ビジネスレイヤ] 参照ペインの上部にある [検索パネルの表示/非表示] アイコン  をクリックします。  
[オブジェクトの検索] パネルがビジネスレイヤの下部に開いて、すべてのオブジェクトが表示されます。
- 3 オブジェクトの種類でフィルタを適用するには、[オブジェクトの検索] パネルで、フィルタアイコン  をクリックします。オブジェクトの種類の一覧から、検索に含める種類を選択します。  
[オブジェクトの検索] パネルに選択した種類のオブジェクトのみが表示されます。
- 4 オブジェクト名でテキスト検索を実行するには、[検索バーの表示/非表示] アイコン  を選択して、検索するテキストを入力します。  
[オブジェクトの検索] パネルに、入力したテキストが含まれる名前のオブジェクトのみが表示されます。
- 5 [オブジェクトの検索] パネルでオブジェクト名をクリックすると、[編集] ペインにオブジェクトのプロパティが開きます。
- 6 検索が終了したら、[検索パネルの表示/非表示] アイコン  を再度クリックして、[オブジェクトの検索] パネルを非表示にします。



## 7.5.6 ビジネスレイヤのツリービューの表示オプションの変更

ビジネスレイヤの編集時には、[ビジネスレイヤ] 参照ペインにビジネスレイヤ内のオブジェクトのツリービューが表示されます。この手順に従って、ビジネスレイヤオブジェクトの表示モードを変更します。

- 1 [ビジネスレイヤ] 参照ペインの上部にある [表示オプション] アイコン  をクリックします。
- 2 OLAP 接続に基づくビジネスレイヤの場合、以下の 3 つのオプションから 1 つを選択します。
  - ・ [名前の表示] では、ビジネスレイヤオブジェクトのプロパティに割り当てられたオブジェクト名を表示します。
  - ・ [技術名の表示] では、キューブからオブジェクト名を表示します。
  - ・ 名前と技術名の両方を表示
- 3 ビジネスレイヤのツリービューにアクティブなオブジェクトのみを表示するには、[アクティブでないオブジェクトの非表示] を選択します。

表示オプションは、エディタを閉じるまで有効なままです。

## 7.5.7 オブジェクトのキーの定義



オブジェクトのキーの定義は、データファンデーションで作成されたディメンションでのみ使用できます。

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤ名をダブルクリックします。
- 2 [ビジネスレイヤ] ペインでディメンションを選択します。
- 3 編集ペインで、[キー] タブをクリックします。
- 4 [キーの追加] をクリックします。
 

1 つのディメンションに対して 1 つの 1 次キーと複数の外部キーを定義できます。最初に追加されるキーが 1 次キーです。

### 注

データベース内の既存のキー列を検出するには、[検出] をクリックします。

- 5 テーブルでキーを選択し、[SELECT] 列をクリックして SQL SELECT 句を追加します。  
SQL エディタで文を構築するには、[SELECT] 列の最後にある  を選択します。
- 6 [WHERE] 列をクリックして SQL WHERE 句を追加します。  
SQL エディタで文を構築するには、[WHERE] 列の最後にある  を選択します。
- 7 [アクティブ] 列でクリックし、キーを有効化または無効化します。

### 関連項目

- ・ 116 ページの [キービジネスレイヤオブジェクトのプロパティ](#)

## 7.6 ビジネスレイヤビューについて

[ビジネスレイヤ] ペインに表示するオブジェクト数を制限するビジネスレイヤビューを使用して、ビジネスレイヤオブジェクトの表示を編集することができます。ビジネス関係を共有するオブジェクトをグループ化するために、ビジネスレイヤビューを使用します。


ビジネスレイヤビューはクエリパネルで選択できます。ビジネスレイヤビューを使用して、特定のユーザまたはグループに対して、ビジネスレイヤオブジェクトの使用を許可または拒否するために、セキュリティを定義できます。ビジネスレイヤビューを使用したセキュリティの定義についての詳細は、「ビジネスセキュリティプロファイルのクエリの作成設定」の関連項目を参照してください。

ビジネスレイヤビューからエディタ内の [ビジネスレイヤ] ペインにフィルタを適用することもできます。

### 関連項目

- ・ 122 ページの[ビジネスレイヤビューの作成と編集](#)
- ・ 123 ページの[ビジネスレイヤビューによるフィルタの適用](#)
- ・ 187 ページの[ビジネスセキュリティプロファイルのクエリの作成設定](#)

### 7.6.1 ビジネスレイヤビューの作成と編集

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤ名をダブルクリックします。
- 2 [ビジネスレイヤ] ペインの上部にある [ビジネスレイヤビューの管理] アイコン  をクリックします。  
[ビジネスレイヤビューの編集] ダイアログボックスが開きます。
- 3 次のいずれかを実行します。
  - ・ ビューを追加するには、[追加] をクリックします。
  - ・ 既存のビューを編集するには、一覧からビューを選択します。

#### 注

ただし、[マスタ] ビューを編集することはできません。

- 4 [名前] テキストボックスでビューの名前を編集します。
- 5 [ビューのオブジェクト] ボックスで、ビジネスレイヤのオブジェクトの横にあるチェックボックスを選択または選択解除して、これらをビューに含めるか除外します。  
すでにビューに含まれているオブジェクトでのみ作業を行うには、[選択されたオブジェクトのみを表示] を選択します。
- 6 [説明] テキストボックスにビューの説明を入力するか編集します。

## 7.6.2 ビジネスレイヤビューによるフィルタの適用

デフォルトでは、ビジネスレイヤ内のすべてのフォルダおよびオブジェクトが、エディタ内の [ビジネスレイヤ] ペインに表示されます。ビジネスレイヤビューを使用して、[ビジネスレイヤ] ペインでの表示項目にフィルタを適用することができます。

少なくとも 1 つのビジネスレイヤビューを定義する必要があります。

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤ名をダブルクリックします。
- 2 [ビジネスレイヤ] ペインの上部にあるドロップダウンリストから、ビジネスレイヤビューを選択します。  
オブジェクトの完全な一覧を返すには、ドロップダウンリストから [マスタ] を選択します。

### 関連項目

- ・ 122 ページの[ビジネスレイヤビューの作成と編集](#)

## 7.7 ビジネスレイヤのクエリオブジェクトについて

クエリオブジェクトは、ビジネスレイヤに保存されて関連付けられたクエリです。クエリオブジェクトは、エディタの [クエリ] ペインでカタログ化され、クエリパネルで作成されます。

### 注


クエリは、インフォメーションデザインツール内で、ビジネスレイヤをテストする、またはクエリをプレビューするために使用できます。クエリオブジェクトは、公開されたユニバーを使用するレポートおよび分析ツール製品では使用できません。

### 関連項目

- ・ 123 ページの[ビジネスレイヤへのクエリオブジェクトの追加](#)
- ・ 137 ページの[クエリパネルについて](#)

### 7.7.1 ビジネスレイヤへのクエリオブジェクトの追加

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤ名をダブルクリックします。
- 2 [ビジネスレイヤ] ペインの下部にある [クエリ] タブをクリックします。

- 3 [クエリ] ペインの上部にある [クエリの挿入] アイコン  をクリックします。  
クエリパネルが開きます。
- 4 クエリパネルで、クエリを構築し、[OK] をクリックします。  
[クエリ] ペインで新しいクエリが使用できます。

#### 関連項目

- ・ 137 ページの[クエリパネルについて](#)

## 7.8 パラメータについて

パラメータは、実行時に値が必要になる、ビジネスレイヤまたはデータファンデーションの変数です。パラメータには 2 つの入力タイプがあります。

- ・ プロンプトに対するユーザ入力。プロンプトは、結果セットを制限するために 1 つ以上の値を設定することをユーザに求める質問または指示です。
- ・ 実行時にパラメータの固定値を指定する定義済みの入力。

パラメータはビジネスレイヤおよびデータファンデーションの個別のコンポーネントとして定義され、ビジネスレイヤのすべてのオブジェクトで使用できます。オブジェクトの SQL または MDX 定義の中でパラメータオブジェクトを使用して、ユーザに入力を求めたり、クエリに対し固定値の応答を返します。

#### 注

データファンデーションに挿入される各種パラメータは、データファンデーションで作成されるすべてのビジネスレイヤが継承します。これらのパラメータはビジネスレイヤ内では編集できません。これらは、データファンデーション内で編集する必要があります。

次のプロパティがパラメータで使用できます。

プロパティ	説明
ユーザへのプロンプト	選択されている場合、ユーザは実行時に値の入力を求められます。 選択されていない場合、定義済みの値が実行時にパラメータに入力されます。
プロンプトテキスト	[ユーザへのプロンプト] が選択されている場合のプロンプトの質問または指示のテキスト。
値を設定	[ユーザへのプロンプト] チェックボックスがオフになっているとき使用できます。実行時にパラメータに使用される 1 つ以上の値を入力できます。
データ型	プロンプトに答えるために必要なデータ型
複数の値を許可	選択されている場合、ユーザが値の一覧から複数の値を選択できます。

プロパティ	説明
最後に選択した値を維持	選択されている場合、ユーザが最後に選択した値がプロンプトの再実行時に維持されます。
インデックス認識プロンプト	選択されている場合、キー列がプロンプトに含まれ、一覧の値が制限されます。キー列はユーザには見えません。
値の一覧の関連付け	プロンプトの値を提供する値の一覧。
一覧から選択	選択されている場合、ユーザは一覧のメンバーを選択することを強制されます。
デフォルト値の設定	デフォルトとして使用される値を選択できます。

#### 関連項目


- ・ 125 ページの[パラメータの挿入と編集](#)」
- ・ 130 ページの[値の一覧のビジネスレイヤで定義済みのプロンプトへの関連付け](#)」

### 7.8.1 パラメータの挿入と編集

パラメータエディタは、ビジネスレイヤまたはデータファンデーションエディタのタブから起動できます。

#### 注

データファンデーションに挿入される各種パラメータは、データファンデーションで作成されるすべてのビジネスレイヤが継承します。これらのパラメータはビジネスレイヤ内では編集できません。これらは、データファンデーション内で編集する必要があります。

- エディタの参照ペインにある [パラメータおよび値の一覧] タブをクリックします。
- 次のいずれかを実行します。
  - パラメータを挿入するには、[パラメータ] ペインの上部にある [パラメータの挿入] アイコン  をクリックします。
  - パラメータを編集するには、一覧でパラメータ名をクリックします。  
パラメータのプロパティが [パラメータ] ペインの右側にあるエディタに表示されます。
- 必要に応じて、プロパティを編集します。パラメータのプロパティの説明は、関連トピックにあります。

#### 関連項目

- ・ 124 ページの[パラメータについて](#)」
- ・ 130 ページの[値の一覧のビジネスレイヤで定義済みのプロンプトへの関連付け](#)」

## 7.9 値の一覧について

値の一覧 (LOV) とは、オブジェクトに関連付けられたデータ値を記録したリストです。LOV を使って、ユーザは、クエリに関連するオブジェクトが含まれたときのプロンプトへの返答としての値を選択できます。LOV を使って、データセットを選択した値に制限できます。

LOV はビジネスレイヤまたはデータファンデーションの個別のコンポーネントであり、ビジネスレイヤのすべてのビジネスオブジェクトで使用できます。LOV は、いつでもオブジェクトと関連付けられます。

### 注

データファンデーションに挿入される LOV は、データファンデーションで作成されるすべてのビジネスレイヤが継承します。これらの LOV はビジネスレイヤ内では編集できません。これらは、データファンデーション内で編集する必要があります。

以下の種類の値の一覧を定義できます。

LOV の種類	説明
ビジネスレイヤオブジェクトに基づく値の一覧 (ビジネスレイヤでのみ使用可能)	LOV は、ビジネスレイヤでオブジェクトを含むクエリまたはカスタム階層のいずれかを基にしています。一覧は、クエリから返された値か、階層値を基にしています。
静的値の一覧	LOV は手動で入力されるかファイルからインポートされる、指定された値の一覧を基にしています。
カスタム SQL に基づく値の一覧	LOV は、指定された SQL 式から返された値を基にしています。

### 関連項目

- ・ 126 ページの[値の一覧のクエリオプション](#)」
- ・ 127 ページの[値の一覧の列のプロパティ](#)」
- ・ 128 ページの[値の一覧の挿入および編集](#)」

### 7.9.1 値の一覧のクエリオプション

値の一覧 (LOV) のプロパティにある [オプション] タブでは、LOV でユーザおよびクエリの制約設定できます。次のオプションがあります。

オプション	説明
ユーザに値の一覧の編集を許可する	選択されている場合、LOVは、デザイナー以外のユーザが編集およびパーソナライズすることができます。
使用前に自動更新	選択されている場合、LOV は、LOV が呼び出されるたびに自動的に最新表示されます。毎回、LOV ファイルが最新表示されるので、パフォーマンスに影響を及ぼします。LOV が大量の値を返す場合、このオプションを無効にしておく必要があります。
使用前にユーザに対して値のフィルタを強制	選択されている場合、この OV を使用してクエリを実行するユーザは、検索条件を入力してから、LOV の値にフィルタを適用する必要があります。検索条件に一致する値のみが、LOV に返されます。一致条件を定義するのに使用される文字列は、以下のとおりです。 <ul style="list-style-type: none"> <li>・ *: ゼロ文字を含む任意の数の文字数に一致</li> <li>・ ?: 1 つの文字に一致</li> <li>・ ¥: 次の文字をエスケープして、ワイルドカード文字を検索</li> </ul>
ユーザにデータベースの値の検索を許可する	選択されている場合、この LOV を使用してクエリを実行するユーザは、データベース内の LOV の値を検索できます。このオプションは、ユーザが部分的な LOV の結果で検索を実行する際に便利です。
クエリ実行のタイムアウト	選択されている場合、LOV クエリの実行時間を秒単位で制限します。
最大行数	選択されている場合、LOV クエリで返される最大行数を入力できます。

#### 関連項目

- ・ 126 ページの[値の一覧について](#)」

## 7.9.2 値の一覧の列のプロパティ

値の一覧 (LOV) のプロパティにある [プロパティ] タブでは、LOV で列のプロパティを編集できます。プロパティのテーブルでプロパティ列をクリックして、以下のプロパティを編集できます。

プロパティ	説明
列の名前	列の名前を編集できます。
キー列	インデックス認識キーとなる列を選択できます。
データ型	列のデータ型を選択できます。
隠す	選択されている場合、列はユーザに対して表示されません。たとえば、別の列のキーとしてのみ使用されている列を非表示にすることができます。

## 関連項目


- ・ 126 ページの[値の一覧について](#)」

### 7.9.3 値の一覧の挿入および編集


値の一覧エディタは、ビジネスレイヤまたはデータファンデーションエディタのタブから起動できます。

**注**

データファンデーションに挿入される値の一覧 (LOV) は、データファンデーションで作成されるすべてのビジネスレイヤが継承します。これらの LOV はビジネスレイヤ内では編集できません。これらは、データファンデーション内で編集する必要があります。

- 1 エディタの参照ペインにある [パラメータおよび値の一覧] タブをクリックします。
- 2 次のいずれかを実行します。
  - ・ LOV を挿入するには、[値の一覧] ペインの上部にある [値の一覧の挿入] アイコン  をクリックして、LOV の種類を選択します。種類については、値の一覧の関連項目で説明しています。
  - ・ LOV を編集するには、一覧で LOV 名をクリックします。  
LOV のプロパティが [値の一覧] ペインの右側にあるエディタに表示されます。
- 3 必要に応じて、プロパティとクエリオプションを編集します。プロパティは LOV の種類によって異なります。



オプション	説明
ビジネスレイヤオブジェクトに基づく値の一覧 (ビジネスレイヤでのみ使用可能)	<p>クエリの LOV の基になります。</p> <ul style="list-style-type: none"> <li>a [定義] タブで、[クエリパネルに基づく値の一覧] を選択します。</li> <li>b [クエリの編集] をクリックします。</li> <li>c クエリパネルでオブジェクトを選択して、クエリフィルタを定義し、必要な値の一覧を返すクエリを定義します。</li> <li>d [OK] をクリックします。</li> </ul> <p>カスタム階層の LOV の基になります</p> <ul style="list-style-type: none"> <li>a [定義] タブで、[カスタム階層に基づく値の一覧] を選択します。</li> <li>b [ディメンションの追加] をクリックします。</li> <li>c LOV に必要な階層を作成するために、一覧からディメンションを選択します。一覧内のディメンションの順序は、階層でのレベルを表します。上矢印および下矢印キーを使用して、順序を変更します。</li> <li>d [OK] をクリックします。</li> </ul> <p>定義済み一覧の値を確認するには、[プレビュー] をクリックします。</p>
静的値の一覧	<p>値を手動で追加する:</p> <ul style="list-style-type: none"> <li>a [定義] タブで、[列の追加] をクリックして列をテーブルに追加します。表に示す列の値を入力します。</li> <li>b 行を追加するには、テーブルの右側にある [行の追加] アイコン  をクリックします。</li> </ul> <p>ファイルから一覧を作成するには、以下のとおりです。</p> <ul style="list-style-type: none"> <li>a [定義] タブで、[インポート] をクリックします。</li> <li>b 静的な一覧の値としてインポートするファイルを、.txt、.csv、.prn、または .asc ファイルから選択します。</li> <li>c ファイル内のデータの種類によって、[データ区切り]、[テキスト区切り]、[日付の形式] オプションを設定します。</li> <li>d [OK] をクリックします。</li> </ul> <p>[プロパティ] タブで列のプロパティを編集できます。列のプロパティの詳細については、関連項目を参照してください。</p>
カスタム SQL に基づく値の一覧	<ul style="list-style-type: none"> <li>a [定義] タブで、[SQL の編集] をクリックします。</li> <li>b SQL エディタで、必要な値を返すための SQL 式を構築し、[OK] をクリックします。</li> </ul> <p>定義済み一覧の値を確認するには、[プレビュー] をクリックします。</p> <p>[プロパティ] タブで列のプロパティを編集できます。列のプロパティの詳細については、関連項目を参照してください。</p>

- 4 [オプション] タブで、LOV のクエリオプションを設定します。クエリオプションの詳細については、関連項目を参照してください。

- 5 ビジネスレイヤまたはデータファンデーションを保存します。


#### 関連項目

- ・ 126 ページの[値の一覧について](#)」
- ・ 127 ページの[値の一覧の列のプロパティ](#)」
- ・ 126 ページの[値の一覧のクエリオプション](#)」

## 7.9.4 値の一覧のビジネスオブジェクトへの関連付け

値の一覧 (LOV) はビジネスレイヤで使用可能である必要があります。LOV はビジネスレイヤエディタの [パラメータおよび値の一覧] タブ内の一覧にあります。



LOV をビジネスオブジェクトに関連付けて、オブジェクトがクエリパネルでフィルタとして使用される場合に入力する可能性のある値を制限します。

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤ名をダブルクリックします。
- 2 [ビジネスレイヤ] タブをクリックして、[ビジネスレイヤ] ペインを開きます。
- 3 [ビジネスレイヤ] ペインでビジネスレイヤオブジェクトをクリックします。
- 4 編集ペインで [詳細] タブをクリックします。
- 5 [値の一覧の関連付け] チェックボックスをオンにします。
- 6 参照アイコン  をクリックして、一覧から LOV を選択し、[OK] をクリックします。
- 7 ビジネスレイヤを保存します。

#### 関連項目


- ・ 128 ページの[値の一覧の挿入および編集](#)」

## 7.9.5 値の一覧のビジネスレイヤで定義済みのプロンプトへの関連付け

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤ名をダブルクリックします。
- 2 [ビジネスレイヤ] ペインの下部にある [パラメータおよび値の一覧] タブをクリックします。
- 3 [パラメータ] ペインにある一覧でパラメータをクリックするか、[パラメータの挿入] アイコン  をクリックして新しいパラメータを定義します。  
パラメータのプロパティが [パラメータ] ペインの右側にあるエディタに表示されます。
- 4 [ユーザへのプロンプト] チェックボックスをオンにします。
- 5 [値の一覧の関連付け] フィールドの最後にある [参照] アイコン  をクリックします。

- 6 LOV の種類をラジオボタンで選択します。

種類	説明
ビジネスオブジェクト LOV	ビジネスレイヤで、オブジェクトから LOV の値を選択します。
ユニバースの値の一覧	定義済みのカスタマイズされた LOV を選択します。これらは、[値の一覧] ペインに一覧表示されている LOV です。

- 7 ビジネスレイヤオブジェクトまたは事前定義済みの LOV を選択し、[OK] をクリックします。
- 8 デフォルト値にするために、リスト内で使用可能な値を制限する場合、[デフォルト値の設定] を選択し、フィールドの最後にある [参照] アイコン  をクリックします。

選択ボックスが表示され、選択したオブジェクトまたはリストに対して使用可能な値が一覧表示されます。左側で値を選択して、[選択した値] 一覧を作成し、[OK] をクリックします。

これで、この手順内で定義されたパラメータの名前のある @Prompt 関数を使用して、ビジネスレイヤ内のオブジェクトの SQL または MDX 定義でプロンプトおよび LOV を含めることができます。: @Prompt(<パラメータ名>)

#### 関連項目

- ・ 128 ページの[値の一覧の挿入および編集](#)」
- ・ 124 ページの[パラメータについて](#)」
- ・ 126 ページの[値の一覧について](#)」
- ・ 116 ページの[ビジネスレイヤオブジェクトのプロパティの詳細設定](#)」

## 7.10 オブジェクトのナビゲーションパスについて

ナビゲーションパスとは、SAP Business Objects レポーティングツールで使用されるドリルパスを定義するオブジェクトのことです。ドリルパスとは、レポートアナリストがディメンションでドリルダウンできるようにするドリル可能なビジネスオブジェクトのリストのことです。

ナビゲーションパスオブジェクトは、次の 2 種類のいずれかになります。


ナビゲーションパスの種類	説明
デフォルト	<p>ビジネスレイヤ内でビジネスオブジェクトの階層構成によって定義されるパスです。ビジネスレイヤに分析ディメンションが含まれる場合、ナビゲーションパスには各分析ディメンションの下にあるディメンションが含まれます。それ以外の場合、ナビゲーションパスには各フォルダ内のディメンションが含まれます。</p> <p>デフォルトのナビゲーションパスは、ビジネスレイヤエディタ内の [ナビゲーションパス] タブで表示できます。デフォルトのパスは編集できません。</p>

ナビゲーションパスの種類	説明
カスタム	使用可能なディメンションに基づくパスを定義します。

## 関連項目

- ・ 132 ページの[ビジネスレイヤへのナビゲーションパスオブジェクトの挿入](#)

### 7.10.1 ビジネスレイヤへのナビゲーションパスオブジェクトの挿入

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト]ビューでビジネスレイヤ名をダブルクリックします。
- 2 [ビジネスレイヤ] ペインの下部にある [ナビゲーションパス] タブをクリックします。
- 3 [ナビゲーションパス] ペインの上部にある [カスタム] を選択します。
- 4 [ナビゲーションパスの挿入] アイコン  をクリックします。
- 5 パスの [名前] と [説明] (オプション) を入力します。  
この名前と説明は、公開されたユニバースを使用するクエリツールおよびレポートツールに表示できます。
- 6 [追加] をクリックして、パスのディメンションを選択します。上下の矢印キーを使用して、一覧内のディメンションの順序を変更します。
- 7 ビジネスレイヤを保存します。

## 関連項目

- ・ 131 ページの[オブジェクトのナビゲーションパスについて](#)

### 7.11 アグリゲート認識について

アグリゲート認識とは、データベース内の集計テーブルを利用するためにユニバースに設定する機能です。オブジェクトの SELECT 句で @Aggregate\_Aware という関数を使うことによって、未集計のデータを含むテーブルではなく、集計テーブルに対してクエリが実行されます。

アグリゲート認識とユニバース内での設定方法の詳細については、『ユニバースデザインツールユーザガイド』を参照してください。

## 関連項目

- ・ 133 ページの[アグリゲート認識の設定](#)

### 7.11.1 アグリゲート認識の設定

- 1 エディタでビジネスレイヤを開くには、[ローカルプロジェクト]ビューでビジネスレイヤ名をダブルクリックします。
- 2 インフォメーションデザインツールのメインメニューで、[アクション] > [集計テーブルの自動参照の設定] を選択します。  
[アグリゲート認識] ダイアログボックスで、オブジェクトを含めるテーブルを指定します。オブジェクトは、集計適用に対して最適化されたオブジェクトを含む集計テーブルとは互換性がありません。
- 3 左のボックスで、集計テーブルをクリックします。
- 4 右のボックスで、各非互換オブジェクトのチェックボックスを選択します。
- 5 データファンデーションの各集計テーブルで上記の手順を繰り返します。
- 6 すべてのテーブルですべての非互換オブジェクトが指定されたら、[OK]をクリックします。

#### 注

このダイアログボックスには、非互換オブジェクトの指定をサポートする[非互換の検出]ボタンがあります。テーブルをクリックしてからこのボタンをクリックすると、非互換オブジェクトが自動的に選択されます。[非互換の検出]で検出された非互換オブジェクトは、あくまでも目安として考え、最終的な判断はユーザ自身で行ってください。

#### 関連項目

- ・ 132 ページの[アグリゲート認識について](#)

### 7.12 ビジネスレイヤの参照について

OLAP キューブに基づくビジネスレイヤの場合、[ビジネスレイヤの最新表示] ウィザードが OLAP キューブで変更を検出し、ビジネスレイヤに変更を適用します。

[オプションの選択] ページで、ウィザードがキューブで検出する必要がある変更の種類を選択できます。

検出に基づいて、ウィザードでは可能な更新アクションが [アクションの選択] ページに一覧表示されます。ビジネスレイヤに適用する更新アクションを選択できます。

変更を適用する前に、ウィザードでは更新アクションの概要が [集計の最新表示] ページに表示されます。概要をファイルに保存できます。ウィザードを終了する前に、戻って、選択を変更することができます。

概要一覧には、キューブ構造での変更に基づいて、ビジネスレイヤに提示される変更が表示されます。更新を適用する前に、提案された変更をクリアおよび選択することができます。

**注**

最新表示は、[元に戻す]アクションを使用して元に戻すことができます。元に戻すと、ビジネスレイヤが最新表示の前の状態に戻ります。元に戻すには、インフォメーションデザインツールのメインメニューで、[編集] > [元に戻す] を選択します。

**関連項目**

- ・ 134 ページの [OLAP キューブに基づくビジネスレイヤの最新表示](#)」

## 7.12.1 OLAP キューブに基づくビジネスレイヤの最新表示

[ビジネスレイヤの最新表示] ウィザードを使用し、ビジネスレイヤの作成後、または前回の最新表示後に行われた OLAP キューブでの変更に基づいてビジネスレイヤを更新します。

- 1 ビジネスレイヤを開くには、[ローカルプロジェクト] ビューでビジネスレイヤ名をクリックします。
- 2 ウィザードを開始するには、インフォメーションデザインツールのメインメニューで、[アクション] > [構造の最新表示] を選択します。
- 3 ウィザードページの指示に従います。特定のページでの操作方法については、ヘルプアイコンをクリックしてください。

**関連項目**

- ・ 133 ページの [ビジネスレイヤの参照について](#)」

## 7.13 最適化されたクエリ実行のための統計の計算について

複数ソースユニバースでは、データフェデレーションエンジンで正確なテーブルと列の統計が使用可能な場合に、最高のパフォーマンスを得られます。データフェデレーションエンジンのコストに基づくオプティマイザでは、これらの統計を使用して、最適な結合メソッドと順番を決定します。

[統計の計算] コマンドで、ユニバースのリポジトリでの統計の計算および保存ができるようになるため、クエリの実行を最適化します。

ボリューム内で変更される可能性のあるテーブルや、列値が頻繁に変更されるテーブルの場合は、定期的に統計を計算する必要があります。

以下の統計が、最適化プロセスのために生成されます。

- ・ テーブルの行数
- ・ 列の固有値の数

次のオプションを設定します。

- ・ 特定日より前に計算されたすべてのテーブルと列の選択
- ・ 一度も計算されなかったすべてのテーブルと列の選択

- ・ 各テーブルと列の選択
- ・ 各テーブルと列の選択解除

#### 関連項目

- ・ 135 ページの[複数ソースユニバースの統計の計算](#)

### 7.13.1 複数ソースユニバースの統計の計算

複数のソースが有効化されたデータファンデーションに基づくユニバースに対してのみ、統計を計算できます。

- 1 以下の手順で、ビジネスレイヤまたは公開されたユニバースから統計を計算できます。
  - ・ リポジトリリソースビューで公開されるユニバースから統計を計算するには、ユニバースが公開されるリポジトリでセッションを開きます。ユニバースを右クリックし、[統計の計算] を選択します。
  - ・ ビジネスレイヤから統計を計算するには、[ローカルプロジェクト] ビューでビジネスレイヤ名をクリックして、エディタでビジネスレイヤを開きます。[ビジネスレイヤ] ペインでビジネスレイヤ名を右クリックし、[統計の計算] を選択します。

[統計の計算] ダイアログボックスが表示されます。

- 2 統計を計算するテーブルと列を選択します。  
テーブルをチェックする場合、このテーブルに対応するすべての列が選択されます。
- 3 [計算] をクリックします。  
統計が計算され、リポジトリに保存されます。大規模なデータベースの場合、処理に数分以上かかる場合があります。計算の実行中に、ウィンドウを閉じて、インフォメーションデザインツールで別のタスクを実行することができます。

#### 関連項目

- ・ 134 ページの[最適化されたクエリ実行のための統計の計算について](#)





## クエリパネルの使用法

### 8.1 クエリパネルについて

[クエリパネル] を使用して、クエリの構築、テスト、プレビューを行います。結果オブジェクトを [結果オブジェクト] 枠に挿入します。これらの結果オブジェクトはクエリレポートに表示され、日付範囲、指定された値より大きい結果、選択した範囲などを条件に返された結果をフィルタするため、フィルタオブジェクトを追加します。クエリによって予想した結果が返されることを確認するため、結果をプレビューできます。また、クエリを構成するクエリ構文も表示できます。

[クエリパネル] を使用すると以下のタイプのクエリを作成できます。

- ・ OLAP ユニバースの階層クエリ
- ・ リレーショナルユニバースの非階層クエリ
- ・ BusinessObjects Enterprise XI 3.X ユニバースの非階層クエリ

#### 関連項目

- ・ 137 ページの[クエリパネルの説明](#)」
- ・ 141 ページの[メンバーセレクトについて](#)」
- ・ 148 ページの[クエリについて](#)」
- ・ 151 ページの[クエリを作成する](#)」
- ・ 151 ページの[クエリにフィルタを追加する](#)」

#### 8.1.1 クエリパネルの説明

クエリパネルは、次の要素で構成されています。

表 8-1: クエリパネルの要素

クエリパネルの要素	説明
ビジネスレイヤのツリービュー (左側)	この枠には、使用可能なクラスとオブジェクトがツリー構造で表示されます。ノードをクリックするとブランチまたは階層が展開され、ノードを再度クリックするとその階層が閉じるか折りたたまれます。表示モード (各オブジェクトのキャプション、技術名、またはその両方) を選択します。この枠でオブジェクトを検索するには、[フィルタ] ボタンをクリックして検索文字列を入力します。クエリを作成するには、オブジェクトをこの枠から [結果オブジェクト] 枠または [クエリフィルタ] 枠にドラッグします。
[クエリの結合] 枠 (左下)	この枠は、クエリを結合しているときにのみ表示されます。結合中のクエリの構造が表示されます。クエリアイコンを移動させて、クエリの結合方法を再構築できます。クエリアイコンをクリックすると、[オブジェクト] 枠と [フィルタ] 枠にクエリのプロパティが表示されます。
[結果オブジェクト] 枠 (右側)	ビジネスレイヤビューから、クエリに含めるオブジェクトを選択して、この枠にドラッグします。これらのオブジェクトは結果レポートの列ヘッダとして返されます。階層オブジェクトをここに配置する場合は、メンバーセレクトツールを使用して表示し、クエリに含める階層のメンバーを選択します。クエリから除外するメンバーを選択することもできます。
[クエリフィルタ] 枠 (右側)	[フィルタパネル (表示/非表示)] をクリックし、オブジェクトをこの枠にドラッグしてクエリを制限することで、結果データを制限できます。たとえば、特定の値や値の範囲を返すように結果を制限できます。
[データプレビュー] 枠 (右側)	[ <input type="checkbox"/> データプレビュー枠の表示/非表示] をクリックすると、この枠を使用してクエリの結果をテストすることができます。表示される結果をプレビューしたり、クエリを変更してその結果をプレビューすることができます。

クエリパネルには次のボタンもあります。

- ・ クエリの結合: リレーショナルユニバース用にクエリを結合する
- ・ クエリのプロパティ: クエリのプロパティを表示および編集する
- ・ スクリプトの表示: クエリの構造を表示するスクリプトを編集するには、[カスタムのクエリスクリプトを使用] を選択し [クエリスクリプト] 枠でスクリプトを編集します。

**注**

XI 3.X ユニバースを選択して、XI 3.X クエリパネルと同じ方法でクエリを作成できます。メンバーセレクトは使用できません。また、階層オブジェクトのようなディメンショナルオブジェクトを、レベルメンバー、集合メンバー、および計算メンバーと共にクエリに含めることはできません。

### 8.1.2 [結果オブジェクト] 枠について

この枠には、レポートの列にヘッダとして表示するオブジェクトが含まれます。この枠にオブジェクトを追加するには、左側のツリービューから [結果オブジェクト] 枠にドラッグアンドドロップします。

### 8.1.3 [クエリフィルタ] 枠について

フィルタ(条件付きステートメント)を使用することで、返される結果を制限できます。必須フィルタまたは定義済みフィルタを使用できます。

必須フィルタは、ユーザがオブジェクト(ディメンション、メジャー、または詳細)を Interactive Analysis のクエリパネルの [結果] 枠に追加したときに起動されます。必須フィルタは、クエリスクリプトには表示されますが、枠には表示されません。

同じクラスに属するオブジェクトが [結果] 枠で選択されていなくても、定義済みのユニバースフィルタをクエリパネルの [クエリフィルタ] 枠に追加します。

必須フィルタが適用されたクラスに属するオブジェクト(ディメンション、メジャー、または詳細)でフィルタを作成します。

#### 8.1.3.1 クエリパネルでのフィルタ追加について

クエリに返されるデータを制限するには、クエリパネルの [フィルタ] 枠に、1 つ以上のフィルタオブジェクトを追加します。次のクエリオブジェクトにフィルタを作成できます。

- ・ 階層
- ・ 階層レベル
- ・ ディメンション
- ・ 属性
- ・ メジャー
- ・ レベル

OLAP キューブからデータをフィルタリングすると、データの評価、集計を行うサブキューブを効率的に作成できます。

#### 注

任意の階層のレポートを参照する場合は、フィルタを使用せずに、[結果オブジェクト] 枠に階層を配置し、メンバーセレクトを使用してクエリを制限します。この方法では、集計されるメジャー値が制限されません。

### 8.1.4 [データプレビュー] 枠について

[クエリパネル] の右下にある[データプレビュー] 枠では、今定義しているクエリの結果を表示できます。クエリ結果は、手動で、または自動的に更新できます (結果はクエリを変更すると更新されます)。次のクエリプロパティを設定できます。

- ・ [最大行数]: クエリプレビューで取得できる最大行数を設定します。これにより、クエリ時間が短縮され、結果に表示されるデータ量が減少します。
- ・ [詳細プレビュー]: クエリ構文および未処理データを表示できる[データ表示] ダイアログが開きます。[未処理データ] 枠では、行または列フィルタを追加してクエリを微調整し、結果を表示できます。これらの変更は実際のクエリでは更新されません。
- ・ [結果セット表示モード]: 結果セットをフラットレイアウトで表示するか、階層レイアウトで表示するか選択できます。

#### 8.1.4.1 [データ表示] ダイアログボックスについて

[データ表示] ダイアログは、[クエリパネル] の [データプレビュー] ペインの [詳細プレビュー] オプションから使用できます。[データ表示] ダイアログを使用して、以下をプレビューします。

- ・ クエリ構文の表示
- ・ 返されたデータを列でフィルタ[カエサレタデータヲレツデフィルタ]
- ・ 返されたデータを行でフィルタ[カエサレタデータヲギョウデフィルタ]
- ・ 最後の変更を元に戻す、またはやり直す
- ・ プレビューしたクエリ結果を .csv または .xml 形式で保存します。

\*フィルタはクエリではなく返されたデータをフィルタします。

この枠ではクエリスクリプトは編集できません。クエリスクリプトを編集するには、[クエリパネル] で [スクリプトの表示] をクリックし、[カスタムのクエリスクリプトを使用] を選択し、[クエリスクリプト] 枠でスクリプトを編集します。

#### 8.1.4.2 クエリ結果をプレビューする

作成中のクエリをテストします。はじめにクエリを保存せずに、クエリのテストを実行できます。

- 1 必要な結果オブジェクトおよびフィルタオブジェクトをすべて追加したかチェックします。
- 2 [結果] 枠上部の [プレビュー] をクリックします。  
クエリの結果が返されます。

返された結果が、そのクエリに期待する内容かどうか検証します。

## 8.1.5 メンバーセレクトアについて

### 8.1.5.1 メンバーセレクトアについて

メンバーセレクトアを使用して OLAP ユニバース階層のメンバーを選択し、次のことを行います。

- ・ ユニバースを作成するときに指定メンバセットを作成します (この機能は [クエリパネル] では使用できません)。
- ・ 階層または階層のメンバーに基づいたクエリを作成する
- ・ クエリから除外するメンバーを定義する

[メンバーセレクトア] は、階層を含むオブジェクトを [結果オブジェクト] 枠でクリックすると表示されます。メンバーセレクトアは次の枠で構成されます。

枠	説明
[メンバーセレクトア] 枠	メンバーセレクトア上部にあり、次の 3 つのタブがある枠です。 <ul style="list-style-type: none"> <li>・ [メンバー] タブ: 階層内固有の関係に従って、メンバを選択、並べ替えまたは除外します。</li> <li>・ [メタデータ] タブ: メタデータを基準に選択または除外します。このタブには、階層レベル、名前付きセット、および計算メンバー別にオブジェクトが表示されます。</li> <li>・ [プロンプト] タブ: プロンプトを作成して、クエリの実行時にユーザがメンバーまたはメタデータを選択できるようにします。</li> </ul>
[概要] 枠	作成中のクエリで選択したメンバーとプロンプト、および除外されたメンバーが表示されます。[概要] 枠に表示される情報は、クエリパネルの [結果オブジェクト] 枠にも表示されます。

#### 関連項目

- ・ 142 ページの[計算メンバーについて](#)
- ・ 142 ページの[名前付きセットについて](#)

- ・ 143 ページの階層の関係でメンバーを選択する」
- ・ 144 ページのレベルでメンバーを選択する」
- ・ 145 ページの計算メンバーを選択する」
- ・ 145 ページの名前付きセットを選択する」
- ・ 146 ページのメンバーを並び替える」
- ・ 146 ページの選択からメンバーまたはメンバーセットを除外する」
- ・ 147 ページのプロンプトを選択に挿入する」
- ・ 148 ページの選択したメンバーのメンバーセクタへの表示」

#### 8.1.5.1.1 階層について

階層とは、関連するオブジェクト（ディメンション）を順番に配列したものです。階層の例の 1 つに “地域” があります。これは、“国”、“地域”、“市町村” などの複数のディメンションがグループ化されています。選択した地域のすべての市、選択した国のすべての市、選択した市の国と地域など、さまざまな視点で階層に関連付けられたデータを観測できます。

#### 8.1.5.1.2 名前付きセットについて

名前付きセットとは、選択してメンバーの個別セットとして保存した、メンバーのグループのことです。通常、複数の名前付きセットが 1 つの階層中に同時に存在することはありませんが、よく使用される複数のクエリやクエリの一部に対応しています。名前付きセットは、エンドユーザがクエリを作成するときにクエリパネルで使用できます。

#### 8.1.5.1.3 計算メンバーについて

計算メンバーとは、データベースで作成する複雑な計算のことです。計算メンバーは、メンバーセクタの [メタデータ] タブで使用できます。

### 8.1.5.2 メンバーの選択について

メンバーセクタを使用して、階層、階層の一部、または OLAP ユニバースの階層メンバーを選択します。次の操作が可能です。

- ・ メンバーのレベルや階層内での関係に従ってメンバーを選択する
- ・ 名前付きセットを選択する
- ・ 計算メンバーを選択する
- ・ クエリから除外するメンバーを指定する
- ・ クエリのメンバーや条件を選択するためのエンドユーザ向けのプロンプトを作成する

クエリで使用するメンバーを定義している場合は、クエリパネルを使用して、フィルタを追加したりそのクエリをプレビューしたりできます。

## 関連項目

- ・ [142 ページの計算メンバーについて](#)
- ・ [142 ページの名前付きセットについて](#)
- ・ [143 ページの階層の関係でメンバーを選択する](#)
- ・ [144 ページのレベルでメンバーを選択する](#)
- ・ [145 ページの計算メンバーを選択する](#)
- ・ [145 ページの名前付きセットを選択する](#)
- ・ [146 ページのメンバーを並び替える](#)
- ・ [146 ページの選択からメンバーまたはメンバーセットを除外する](#)
- ・ [147 ページのプロンプトを選択に挿入する](#)
- ・ [148 ページの選択したメンバーのメンバーセクタへの表示](#)

## 8.1.5.2.1 階層の関係でメンバーを選択する

メンバーセクタのメンバー関係関数を使用して、階層内の関係や位置でメンバーを選択します。選択できる関係は、[メンバーセクタ] 枠でメンバーを選択すると使用可能になります。

**注**

[子]と[子孫] および [親]と[祖先] は、相互に排他的なペアです。1 つのメンバーに、[子]と[子孫]を同時に選択することも、[親]と[祖先]を同時に選択することもできません。

- 1 メンバーセクタの [メンバー] タブで、メンバー名の左側の選択ボックスをクリックします。
- 2 メンバーセクタで選択するメンバーの名前を右クリックします。  
使用可能なオプションの一覧が表示されます。
- 3 次の関係関数から、適切な関係関数を選択します。

オプション	説明
セルフ	選択されたメンバーだけを使用します。これは、デフォルト設定です。
子	親として選択されているメンバーを持つ選択されたメンバーの1レベル下のメンバーを選択します。
子孫	選択されたメンバーのすべての下位レベルにあるすべてのメンバーを選択します (選択済みメンバーを除く)。
親	選択されたメンバーより1レベル上にあり、値の一部を取得するために選択済みメンバーを使用するメンバーを選択します。

オプション	説明
祖先	選択されたメンバーのすべての上位レベルにあるすべてのメンバーを選択します (選択済みメンバーを除く)。
兄弟	選択されたメンバーとして同じ親を持つ同じレベルにあるメンバーを選択します (選択済みメンバーを除く)。
除外する	除外関数 (自己/子/子孫/親/祖先/兄弟) によって指定されたメンバーを除外します。

選択されたメンバーには先頭に fx が付き、[概要] ペインに表示されます。

#### 関連項目

- ・ 142 ページの[計算メンバーについて](#)
- ・ 142 ページの[名前付きセットについて](#)
- ・ 144 ページの[レベルでメンバーを選択する](#)
- ・ 145 ページの[計算メンバーを選択する](#)
- ・ 145 ページの[名前付きセットを選択する](#)
- ・ 146 ページの[メンバーを並び替える](#)
- ・ 146 ページの[選択からメンバーまたはメンバーセットを除外する](#)
- ・ 147 ページの[プロンプトを選択に挿入する](#)

#### 8.1.5.2.2 レベルでメンバーを選択する

選択している階層のレベルすべてに名前が付けられている必要があります。

ある階層の異なるブランチにある、同一レベルのすべてのメンバーを選択できます。たとえば、“時刻” デイメンションですべての四半期を選択したり、“地域” デイメンションで複数の市を選択します。

- 1 階層を [結果オブジェクト] 枠にドラッグします。
- 2 メンバーセレクトを起動します。  
[メンバーセレクト] に階層メンバーがツリー表示されます。
- 3 メンバーセレクトで [メタデータ] タブをクリックします。  
メンバーセレクトに使用可能なレベル、計算メンバー、および名前付きセットが表示されます。
- 4 レベルを選択します。
- 5 [OK] をクリックします。

クエリを実行すると、選択したレベルのメンバーがレポート作成時に動的に計算されます。

#### 関連項目

- ・ 142 ページの[計算メンバーについて](#)
- ・ 142 ページの[名前付きセットについて](#)



- ・ 143 ページの[階層の関係でメンバーを選択する](#)」
- ・ 145 ページの[計算メンバーを選択する](#)」
- ・ 145 ページの[名前付きセットを選択する](#)」
- ・ 146 ページの[メンバーを並び替える](#)」
- ・ 146 ページの[選択からメンバーまたはメンバーセットを除外する](#)」
- ・ 147 ページの[プロンプトを選択に挿入する](#)」

#### 8.1.5.2.3 計算メンバーを選択する

事前に、クエリパネルの [結果オブジェクト] 枠に OLAP ユニバースの階層か階層メンバーをドラッグアンドドロップしておきます。選択した階層には、1 つ以上の計算メンバーが含まれているものとします。

- 1 階層を [結果オブジェクト] 枠にドラッグします。
- 2 メンバーセレクトを起動します。
- 3 メンバーセレクトで [メタデータ] タブをクリックします。  
メンバーセレクトに使用可能なレベル、計算メンバー、および名前付きセットが表示されます。
- 4 計算メンバーを選択します。
- 5 [OK] をクリックします。

##### 関連項目

- ・ 142 ページの[計算メンバーについて](#)」
- ・ 142 ページの[名前付きセットについて](#)」
- ・ 143 ページの[階層の関係でメンバーを選択する](#)」
- ・ 144 ページの[レベルでメンバーを選択する](#)」
- ・ 145 ページの[名前付きセットを選択する](#)」
- ・ 146 ページの[メンバーを並び替える](#)」
- ・ 146 ページの[選択からメンバーまたはメンバーセットを除外する](#)」
- ・ 147 ページの[プロンプトを選択に挿入する](#)」

#### 8.1.5.2.4 名前付きセットを選択する

クエリパネルの [結果オブジェクト] 枠に配置した階層オブジェクトには、1 つ以上の名前付きセットが含まれているものとします。

- 1 階層を [結果オブジェクト] 枠にドラッグします。
- 2 メンバーセレクトを起動します。
- 3 メンバーセレクトで [メタデータ] タブをクリックします。  
メンバーセレクトに使用可能なレベル、計算メンバー、および名前付きセットが表示されます。
- 4 名前付きセットを選択します。
- 5 [OK] をクリックします。

##### 関連項目

- ・ 142 ページの[計算メンバーについて](#)」

- ・ 142 ページの[名前付きセットについて](#)」
- ・ 143 ページの[階層の関係でメンバーを選択する](#)」
- ・ 144 ページの[レベルでメンバーを選択する](#)」
- ・ 145 ページの[計算メンバーを選択する](#)」
- ・ 146 ページの[メンバーを並び替える](#)」
- ・ 146 ページの[選択からメンバーまたはメンバーセットを除外する](#)」
- ・ 147 ページの[プロンプトを選択に挿入する](#)」

#### 8.1.5.2.5 メンバーを並び替える

デフォルトでは、選択されたメンバーは並び替えられておらず、データベースに保存されている順序で表示されます。一覧を文字順に昇順または降順で並び替えることができます。これは、クエリに使用される順序です。

- 1 メンバーの一覧をクリックします。
- 2 [並び替え] ツールバーボタンをクリックして、並び替え順序を入れ替えます。  
並び替えはローカルで実行され、メンバーセレクトに表示されます。

##### 注

並び替えられた一覧はデータベース内の順序には対応しません。並び替えはローカルに実行されます。

##### 関連項目

- ・ 142 ページの[計算メンバーについて](#)」
- ・ 142 ページの[名前付きセットについて](#)」
- ・ 143 ページの[階層の関係でメンバーを選択する](#)」
- ・ 144 ページの[レベルでメンバーを選択する](#)」
- ・ 145 ページの[計算メンバーを選択する](#)」
- ・ 145 ページの[名前付きセットを選択する](#)」
- ・ 146 ページの[選択からメンバーまたはメンバーセットを除外する](#)」
- ・ 147 ページの[プロンプトを選択に挿入する](#)」

#### 8.1.5.2.6 選択からメンバーまたはメンバーセットを除外する

事前に、クエリパネルの [結果オブジェクト] 枠に OLAP ユニバースの階層か階層メンバーをドラッグアンドドロップしておきます。クエリから 1 つ以上のメンバーを除外します。

結果を制限するために、クエリからメンバーを除外できます。メンバーセレクトの [概要] 枠の [除外] 機能を使用すると、クエリに出現させないメンバーまたはメンバーセットを正しく定義できます。たとえば、地域のすべての州の売上げ高を返すクエリから、ある町を除外することができます。次のものを除外できます。

- ・ 明示的なメンバー
- ・ メンバー関数から得られる暗黙的なメンバー
- ・ 階層レベルから得られる暗黙的なメンバー

期待する結果をプレビューすることはできません。結果を確認するためには、レポートを作成する必要があります。クエリの結果から除外するメンバーを選択するには、次の手順を実行します。

- 1 [メンバーセレクト] を起動するには [結果オブジェクト] 枠で階層をクリックします。

- 2 メンバーセクタで適切な機能を使用して、除外するメンバーを定義します。  
定義されたメンバーが[概要] 枠に表示されます。
- 3 除外するメンバーを選択します。
- 4 [除外する] チェックボックスをクリックします。  
除外されたメンバーの名前が[概要] 枠 (および[結果オブジェクト] 枠) に表示されます。名前に線が表示され、クエリから除外されたことを示します。

#### 関連項目

- ・ 142 ページの[計算メンバーについて](#)
- ・ 142 ページの[名前付きセットについて](#)
- ・ 143 ページの[階層の関係でメンバーを選択する](#)
- ・ 144 ページの[レベルでメンバーを選択する](#)
- ・ 145 ページの[計算メンバーを選択する](#)
- ・ 145 ページの[名前付きセットを選択する](#)
- ・ 146 ページの[メンバーを並び替える](#)
- ・ 147 ページの[プロンプトを選択に挿入する](#)

#### 8.1.5.2.7 プロンプトを選択に挿入する


選択しているメンバーが結果オブジェクトまたはフィルタオブジェクトとしてクエリで使用されると、プロンプトが表示されます。ユーザにクエリのメンバーの選択を求めることができます。

- 1 [クエリパネル] で階層オブジェクトの展開ボタンをクリックします。  
[メンバーセクタ] が表示されます。
- 2 [プロンプト] をクリックします。
- 3 [パラメータの有効化] をクリックします。
- 4 必要に応じて、プロンプトテキストを編集します。
- 5 デフォルト値を使用するには、[デフォルト値の設定] をクリックして [編集] をクリックします。  
[プロンプト] ダイアログエディタが表示されます。
- 6 必要に応じて、プロンプトと値を編集します。

#### 関連項目

- ・ 142 ページの[計算メンバーについて](#)
- ・ 142 ページの[名前付きセットについて](#)
- ・ 143 ページの[階層の関係でメンバーを選択する](#)
- ・ 144 ページの[レベルでメンバーを選択する](#)
- ・ 145 ページの[計算メンバーを選択する](#)
- ・ 145 ページの[名前付きセットを選択する](#)
- ・ 146 ページの[メンバーを並び替える](#)
- ・ 146 ページの[選択からメンバーまたはメンバーセットを除外する](#)

#### 8.1.5.2.8 選択したメンバーのメンバーセクタへの表示

メンバーセクタで、[ツリーを展開して選択内容を表示]アイコン  をクリックして、選択したメンバーをツリービューに表示できます。

ツリービューが自動的に展開され、次のメンバーが表示されます。

- ・ 明示的に選択されたメンバー。
- ・ 関連付けられたメンバーの選択に使用されるメンバー。暗黙的に選択された関連付けられたメンバーは必ずしも表示されません。たとえば、フランスというメンバーがその子を選択するのに使用された場合、ツリービューは展開されてフランスを表示します。ノードのフランスに明示的に選択されたメンバーが含まれていない場合、暗黙的に選択された子を表示するノードは展開されません。

#### ヒント

[ツリーを展開して選択内容を表示] コマンドは、すでに展開されているノードは展開しません。表示の長さを短縮するには、ツリービューですべての開いているノードを閉じてからアイコンをクリックします。

## 8.2 クエリについて

クエリパネルを使用してクエリを作成します。単一のクエリを作成することも、結合クエリを作成することもできます。階層のメンバーであるオブジェクトを追加する場合は、メンバーセクタを使用して、階層のどのメンバーを次の目的で使用するか指定します。

- ・ クエリで使用する
- ・ クエリから明示的に除外する

作成中のクエリの結果をプレビューできます。また、クエリの SQL の流れを表示できます。

#### 注

クエリは、最大 15 桁の数値を返します。

### 8.2.1 クエリのプロパティについて

クエリには次のプロパティを設定できます。

プロパティ	説明
名前	クエリに意味のある名前を入力します。同一の名前を別のクエリに使用することはできません。
ユニバース	クエリに関連付けるユニバースを定義します。
重複する行を無視しない	このオプションを選択すると、クエリは重複する行があっても関連する行をすべて返します。重複する行を結果セットから除外する場合は、このオプションを選択解除します。
空白行の取得 (OLAP ユニバースでのみサポート)	通常空白行が発生するのは、複数ディメンションのクエリで 2 つ以上のディメンションのインターセクションにデータが存在しない場合です。 このオプションを選択すると、結果セットに空白セルを含められる行も含まれます。 このオプションを選択しないと、結果セットは空白でないセルを含む行だけで構成されます。
最大受信時間	クエリを停止するまでの最大実行時間を秒単位で定義します。デフォルトでは、この値はユニバースの [実行時間制限] パラメータと同じです。 この値を 0 に設定すると、このオプションは無効になります。 [実行時間制限] パラメータがこの設定値より小さい場合、[実行時間制限] の値がクエリの実行時間を制限するのに使用されます。
最大受信行数	クエリの実行時に表示されるデータの最大行数を定義します。クエリは該当する行をすべて受信しますが、はじめの n 行だけが表示されます。n は、このパラメータで設定する最大行数です。 システム管理者は、ユーザセキュリティプロファイル設定でこの設定を上書きできます。
サンプルの結果セット	データベースでサポートされている場合、このパラメータは n データベース行をサンプリングします。n は、[サンプルの結果セット] の設定値です。この方法は、[最大受信行数] パラメータよりも高速に処理できます。

プロパティ	説明
最新表示時にコンテキストをリセット	<p>これはリレーショナルユニバースでのみ使用できます。このオプションを選択すると、コンテキストを含むクエリを最新表示するときに、そのコンテキストを選択する必要があります。[コンテキストの削除]をクリックすると、直前に選択したコンテキストを削除できます。</p> <p>このオプションを選択しないと、クエリは元のコンテキストを使用して最新表示されます。最後にクエリを実行した後でコンテキストを編集した場合は、そのクエリは新しいクエリと認識されるので、コンテキストを再度選択する必要があります。</p>
ほかのユーザによるクエリ編集を許可 (Web Intelligence のみ)	このオプションを選択すると、他のユーザもクエリビューにアクセスし、ドキュメント内のクエリを変更できるようになります。このオプションを選択しないと、ドキュメントを変更できるのはレポートの作成者だけになります。このオプションは、ドキュメント内のすべてのクエリに適用されます。
プロンプトの順序	クエリに複数のプロンプトがある場合、この機能を使用してそれらのプロンプトがクエリの中で実行される順序を設定します。プロンプトをクリックし、上向きまたは下向きの矢印を使用して、プロンプトの位置を変更します。

### 8.2.1.1 クエリのプロパティ設定を定義する

- 1 クエリをダブルクリックし、[クエリパネル]を開きます。  
[クエリパネル]に、クエリのオブジェクトが表示されます。
- 2 [クエリのプロパティ] ツールバーボタンをクリックします。  
[クエリのプロパティ] ダイアログに、現在のクエリのプロパティが表示されます。
- 3 必要に応じて、クエリのプロパティを編集します。
- 4 [OK] をクリックして [クエリのプロパティ] ダイアログを閉じ、変更を保存します。

### 8.2.2 作成可能なクエリのタイプについて

クエリパネルを使用すると、次のタイプのクエリを作成してテストできます。

- ・ 階層ユニバースの単一（非階層）クエリ。ある国のすべての町など、あるディメンションの同一レベルにあるオブジェクトすべて
- ・ 指定メンバセットを使用するクエリ
- ・ 計算メンバーを使用するクエリ

### 8.2.2.1 クエリを作成する

事前に、使用できるデータソースに基づいたドキュメントを開くか作成します。

クエリパネルでは、[結果オブジェクト] 枠にオブジェクトをドラッグすると、クエリを作成できます。これらのオブジェクトは、フィルタオブジェクトと同様に、レポートデータを返すクエリになります。

- 1 [クエリ] タブをクリックして、クエリパネルを開きます。
- 2 [オブジェクト] 枠から [結果オブジェクト] 枠に、オブジェクトまたはメジャーをドラッグします。
- 3 適切なフィルタオブジェクトを [フィルタオブジェクト] 枠にドラッグします。
- 4 クエリをプレビューします。  
結果が期待どおりかどうか検証します。
- 5 クエリを保存します。  
クエリはドキュメントと共に保存され、最後のユーザはそのクエリをドキュメント内で使用してレポートを作成できます。

### 8.2.2.2 クエリにフィルタを追加する

クエリにフィルタを追加して、返されるデータを制限します。たとえばクエリは、売上げ高や地域コードなどの一連の値や、特定の期間などのデータを返すことができます。

#### 注

[プロンプト] をクリックしたり、プロンプトエディタを使用して設定を定義して、フィルタにプロンプトを追加できます。

- 1 クエリをダブルクリックし、[クエリパネル] を開きます。
- 2 返されるデータを制限するためにフィルタに使用するオブジェクトを選択して、そのオブジェクトを [フィルタ] 枠にドラッグアンドドロップします。
- 3 クエリにプロンプトを追加するには、[フィルタ] 枠の中のオブジェクトをダブルクリックして、メンバーセレクトの [プロンプト] をクリックします。  
プロンプトを挿入すると、[メンバー] 枠と [メタデータ] 枠が無効になります。いずれの設定も変更できません。

### 8.2.2.3 複合クエリを作成する

リレーショナルデータベース用にクエリを複合できます。このオプションを使用する場合は、少なくとも 2 つのクエリが利用可能である必要があります。

#### 注

この機能は、OLAP データベースでは使用できません。

- 1 [クエリの結合] ボタンをクリックして、[複合クエリ] パネルを開きます。  
[複合クエリ] パネルは、クエリパネルの左下に表示されます。このパネルには、2 つのクエリボタンと AND 演算子が表示されます。
- 2 演算子ボタンをダブルクリックして、別の結合演算子に切り替えます。
- 3 [複合クエリ] 枠で片方のクエリ (たとえば [クエリ 1]) をクリックして、クエリオブジェクトを表示します。  
選択されたクエリのオブジェクトが [結果オブジェクト] 枠、[フィルタオブジェクト] 枠、および [プレビュー] 枠に表示されます。
- 4 選択したクエリを編集します。
- 5 2 番目のクエリ (この場合は [クエリ 2]) をクリックして、2 番目のクエリのオブジェクトをクエリパネルに表示します。  
2 番目のクエリのプロパティが表示されます。
- 6 [クエリの結合] ボタンをクリックして、他のクエリを複合クエリに追加します。  
複合クエリの構造は、[複合クエリ] 枠に表示されます。この枠内でクエリをドラッグアンドドロップすると、複合クエリをさらに構築することができます。

### 8.2.2.4 クエリスクリプトを編集する

- 1 クエリパネルで、[スクリプトの表示] をクリックします。  
[クエリスクリプトの表示] ダイアログボックスが表示されます。そのままでは、クエリスクリプトを編集できません。
- 2 [カスタムのクエリスクリプトを使用] をクリックします。  
これで、クエリスクリプトを編集できます。
- 3 [クエリスクリプト] 枠でクエリを編集します。
- 4 [確認] をクリックしてスクリプト構文を確認します。
- 5 [OK] をクリックして変更を保存します。

### 8.2.2.5 名前付きセットおよび計算メンバーを使用するクエリの作成

名前付きセットは、MDX 式で定義したメンバーのコレクション (ネイティブメンバーセット)、またはメンバーセレクトクを使用して定義したメンバーのコレクション (ビジネスメンバーセット) です。オブジェクトを選択するのと同じ方法で、クエリに名前付きセットを選択できます。



### 8.2.3 プロンプトの使用

プロンプトは、クエリオブジェクトにもフィルタオブジェクトにも追加できます。編集ダイアログを使用してプロンプトのデフォルト値を編集するか選択して、デフォルト値を設定できます。そのクエリの直前の実行で最後に選択された値が、プロンプトに提案されるようにする場合は、[最後に選択した値を維持]を選択します。初めてクエリを使用するときは、デフォルト値が設定されているのであれば、その値が提案されます。



## 整合性のチェック

### 9.1 [整合性のチェック] について

[整合性のチェック] 機能を使用して、たとえば、データファンデーション、ビジネスレイヤ、パラメータ、および値の一覧などのユニバースまたはその要素をさまざまな側面から検証します。SQL および MDX 式の整合性およびデザインの制限への準拠をチェックする、事前定義済みルールを選択します。整合性のチェックを実行することにより、公開されたユニバースでクエリおよびレポートを実行したときの問題を回避できるようになります。

[整合性のチェック] ダイアログボックスには、チェックに含まれるルールが表示されます。ルールの一覧は、実行する整合性のチェックを実行するオブジェクトによって変わります。

ルールを選択したら、[整合性のチェック] をクリックします。整合性のチェックが始まり、[整合性のチェック] ダイアログボックスに結果が表示されます。結果をファイルにエクスポートできます。

[整合性のチェック] ダイアログボックスを閉じた後、整合性のチェックの結果は[整合性エラーのチェック]ビューに表示できます。このビューでは、結果をクリックすると適切なエディタに直接移動して問題を修正することができます。

また、リソースを保存するたびに自動的に整合性のチェックを実行するように、バックグラウンドでの整合性のチェックを設定することもできます。

#### 関連項目

- ・ 155 ページの[整合性のチェックの実行](#)
- ・ 156 ページの[整合性エラーのチェックのレビュー](#)
- ・ 157 ページの[ルールの整合性のチェックについて](#)

### 9.2 整合性のチェックの実行

整合性のチェックは、インフォメーションデザインツールで異なるオブジェクトおよびリソースに対して実行できます。

- ・ [ローカルプロジェクト] ビューのリソース (データファンデーション、ビジネスレイヤ、接続およびショートカット)
- ・ [リポジトリリソース] ビューで公開されたユニバース
- ・ エディタ内のデータファンデーションおよびビジネスレイヤの要素 (テーブル、コンテキスト、ビジネスレイヤオブジェクト、クエリ、パラメータ、値の一覧)

- 1 整合性のチェックを実行するリソースまたはオブジェクトを右クリックし、[整合性のチェック] を選択します。
- 2 [整合性のチェック] ダイアログボックスの左側のペインで、適用するルールを選択します。
- 3 [整合性のチェック] をクリックします。  
整合性のチェックの結果が、[整合性のチェック] ダイアログボックスの右側のペインに一覧表示されます。
- 4 結果をテキストファイルに保存するには、[エクスポート] をクリックします。
- 5 結果の確認が完了したら、[OK] をクリックします。

整合性のチェックの結果は、次回整合性のチェックを実行するまで、[整合性エラーのチェック] ビューで確認することができます。詳細については、関連項目を参照してください。

#### 関連項目

- ・ 157 ページの[ルールの整合性のチェックについて](#)
- ・ 156 ページの[整合性エラーのチェックのレビュー](#)

## 9.3 整合性エラーのチェックのレビュー

- 1 インフォメーションデザインツールのメインメニューで、[ウィンドウ] > [整合性エラーのチェック] を選択します。  
[整合性エラーのチェック] ビューでは最新の整合性のチェック結果を一覧にして表示します。
- 2 問題を修正するには、一覧で結果をダブルクリックします。  
結果に関連するオブジェクトがエディタで開きます。たとえば、結果が Customer テーブルの問題に関連する場合、データファンデーションエディタで Customer テーブルが強調表示された状態で開きます。

結果の一覧は、ビューを閉じるか、次回整合性のチェックを実行するまで、整合性エラーのチェック内に保持されます。

#### 関連項目

- ・ 157 ページの[ルールの整合性のチェックについて](#)

## 9.4 バックグラウンドでの整合性のチェックの設定

- 1 インフォメーションデザインツールのメインメニューで、[ウィンドウ] > [基本設定] > [インフォメーションデザインツール] > [整合性のチェック] を選択します。
- 2 [保存時のバックグラウンドでの整合性チェックの有効化] をオンにします。
- 3 バックグラウンドチェックに含めるルールを選択します。
- 4 [適用] をクリックし、[OK] をクリックします。

バックグラウンドでのチェックは、直ちに適用されます。

## 9.5 ルールの整合性のチェックについて

整合性のチェックが完了すると、ルールチェックの結果は、以下の 3 つの重大度のいずれかに当てはまる可能性があります。

重要度	説明
エラー	チェックにより、動作しない個所が検出されました。問題を解決する必要があります。
警告	見つからないオブジェクトについての警告 (たとえば、見つからないキーや見つからないリンク) です。
情報	チェックは OK でした。緑のチェックマークがルールの横に表示されます。

重大度がエラーまたは警告の結果の場合、結果には検出された問題の説明、問題が見つかったリソース、およびプロジェクトが表示されます。

### 関連項目

- ・ 158 ページの[データファンデーションのルールの整合性のチェック](#)」
- ・ 157 ページの[複数ソースユニバースのルールの整合性のチェック](#)」
- ・ 161 ページの[パラメータのルールの整合性のチェック](#)」
- ・ 159 ページの[値の一覧 \(LOV\) のルールの整合性のチェック](#)」
- ・ 162 ページの[データアクセスセキュリティプロファイルのルールの整合性のチェック](#)」
- ・ 163 ページの[OLAP ユニバースのルールの整合性のチェック](#)」
- ・ 165 ページの[OLAP ユニバース内の MDX オブジェクトのルールの整合性のチェック](#)」

### 9.5.1 複数ソースユニバースのルールの整合性のチェック

このチェックでは、次のことを検証します。

- ・ SQL-92 式が SQL-92 構文に準拠していること。
- ・ データ固有の SQL 式が (計算列、派生テーブル) が特定のデータベースの構文に準拠していること。

SQL 式は SQL-92 BNF 文法に基づいて解析されます。エラーの形式は、次のとおりです。

```
Encountered "{0}" at line {1}, column {2}.
Was expecting one of:
{3}
```

関数が不明な場合、エラーメッセージは次のようになります。

Unknown function at line {1}, column {2}

パーサの制限があるため、エラーはこれ以上正確にできません。

データベース固有のデータファンデーションでは、結合、計算列、および派生テーブルの SQL 式の有効性をチェックするために、ルールがすでに存在しています。

表 9-3: データベース固有ルール

ルール名	説明	重要度
結合のチェック	SQL の有効性は、データファンデーションクエリサーバに対してチェックされます。	エラー
計算列のチェック	SQL の有効性は、ソースから独立した計算列の場合、データファンデーションクエリサーバに対してチェックされ、データベース固有の計算列の場合、適切なデータベースに対してチェックされます。	エラー
派生テーブルのチェック	SQL の有効性は、ソースから独立した派生テーブルの場合、DF クエリサーバに対してチェックされ、データベース固有の派生テーブルの場合、適切なデータベースに対してチェックされます。	エラー

## 9.5.2 データファンデーションのルールの整合性のチェック

整合性のチェックを使用して、データファンデーション内の次のルールをチェックできます。

ルール名	説明	重要度
接続のチェック	接続の整合性をチェックします。	エラー
テーブル構造のチェック	データファンデーションのテーブル構造が物理テーブルと同じかどうかをチェックします。	エラー

ルール名	説明	重要度
テーブルのプライマリキーのチェック	テーブルにプライマリキーが含まれていることをチェックします。	警告
孤立テーブルのチェック	テーブルが孤立していないことをチェックします。	警告
結合 SQL のチェック	結合の形式が正しいことをチェックします。	エラー
カーディナリティ定義のチェック	結合のカーディナリティが定義されていることをチェックします。	エラー
検出されたのと同じカーディナリティのチェック	結合のカーディナリティが検出されたカーディナリティと同じであることをチェックします。	警告
コンテキストのチェック	コンテキストが対象とするすべてのループに対して適切であることをチェックします。	エラー
エイリアステーブルのチェック	エイリアステーブルの整合性をチェックします。	エラー
派生テーブルのチェック	派生テーブルの整合性をチェックします。	エラー
計算列 SQL のチェック	計算列が正しく定義されていることをチェックします。	エラー
計算列のデータ型のチェック	計算列のデータ型の整合性をチェックします。	エラー

**注**

データ量が多いと、[カーディナリティのチェック] オプションの実行に時間がかかる場合があります。データが曖昧、または不足していると、不正確な結果を得ることになります。データベースが大きいか、不完全なデータエントリが含まれている可能性がある場合、このオプションは選択しないでください。

### 9.5.3 値の一覧 (LOV) のルールの整合性のチェック

整合性のチェックを使用して、LOV の次のルールをチェックできます。

ルール名	説明	重要度
使用前に静的 LOV のチェック	[使用前にフィルタを適用] オプションが FALSE に設定されていることをチェックします。	警告
静的 LOV の自動更新のチェック	[使用前に自動更新] オプションが TRUE に設定されていることをチェックします。	警告
静的 LOV のユーザがデータベースで値を検索できないことのチェック	[ユーザにデータベースの値の検索を許可する] オプションが FALSE に設定されていることをチェックします。	警告
静的 LOV のタイムアウトの無効化のチェック	[クエリのタイムアウト] オプションが無効化されていることをチェックします。	警告
静的 LOV の最大行数のチェック	[最大行数] オプションが無効化されていることをチェックします。	警告
Business Objects クエリの LOV におけるビジネスアイテムのチェック	現在の LOV が依存する各ビジネスアイテムをチェックします。	警告
Business Objects クエリの LOV におけるコンテキストのチェック	リレーショナルユニバースの場合、クエリの LOV にコンテキストが含まれるかどうかをチェックします。 [はい] の場合、コンテキストが LOV で確定されているかどうかをチェックします。	警告
Business Objects に基づくカスケード LOV のチェック	現在の LOV によって、使用されているビジネスアイテムをチェックします。	警告
カスタム SQL に基づく LOV が空ではないことのチェック	SQL スクリプトが空ではないことをチェックします。	警告
カスタム SQL に基づく LOV の接続のチェック	LOV 接続および LOV データ構造体を利用可能であることをチェックします。	警告
カスタム SQL に基づく LOV のフィルタのチェック	[使用前にフィルタを適用] オプションが無効化されていることをチェックします。	警告
カスタム SQL に基づく LOV ユーザがデータベースを検索できないことのチェック	[ユーザにデータベースの値の検索を許可する] オプションが無効化されていることをチェックします。	警告



ルール名	説明	重要度
表形式 LOV の割り当て条件のチェック	表形式 LOV がディメンション、詳細、およびメジャーにのみ割り当てられることをチェックします。	警告
カスケード LOV 割り当てのチェック	カスケード LOV がディメンションにのみ割り当てられることをチェックします。	警告
階層 LOV の割り当ておよび継承のチェック	階層 LOV(階層に基づく)が階層にのみ割り当てられることをチェックします。階層レベルが、階層の親に関連付けられている LOV に自動的に継承されることをチェックします。	警告

#### 9.5.4 パラメータのルールの整合性のチェック

整合性のチェックを使用して、パラメータの次のルールをチェックできます。

ルール名	説明	重要度
パラメータ名のチェック	パラメータ名が空ではないことをチェックします。	エラー
パラメータのデータタイプのチェック	パラメータのデータタイプが正しく設定されていることをチェックします。	エラー
パラメータの選択タイプのチェック	パラメータタイプが正しく設定されていることをチェックします (mono または multi)。	エラー
プロンプトテキストのチェック	[ユーザに問い合わせる] オプションがオンの場合、プロンプトテキストが空ではないことをチェックします。プロンプトテキストが空の場合、パラメータ名で置換します。	エラー
[ユーザに問い合わせる] のデフォルト値のチェック	[ユーザに問い合わせる] オプションがオンの場合、デフォルト値が空ではないことをチェックします。	エラー

ルール名	説明	重要度
関連する LOV のチェック	関連付する値の一覧 (LOV) が有効なことをチェックします。	エラー

デザイン時に、LOV に関連付けられていないパラメータを定義することができます。パラメータの整合性のチェックでは、パラメータの LOV が無いオブジェクトを、オブジェクトが手動での入力を許可する場合受け入れますが、それ以外の場合、エラーが返されます。

### 9.5.5 データアクセスセキュリティプロファイルのルールの整合性のチェック

以下のルールは、定義されているデータセキュリティプロファイルとユニバース間の不一致を検出するために使用されます。

ルール名	説明	重要度
プロファイルの存在	データセキュリティプロファイルで参照される接続が存在している必要があります。	エラー
プロファイルがリレーショナル	データセキュリティプロファイルで参照される接続が、リレーショナル接続である必要があります。	エラー
プロファイルが OLAP	ビジネスセキュリティプロファイルで参照される接続が、OLAP 接続である必要があります。	エラー
行制限のチェック	行制限定義内の構文をチェックします。	エラー
テーブルマッピングのチェック	データセキュリティプロファイルのテーブルマッピング: テーブルが存在している必要があります。	警告
ビジネスレイヤでのビューの存在のチェック	クエリビジネスセキュリティプロファイルの作成: ビジネスレイヤにはビューがすでに存在しません。	警告
ビジネスレイヤにオブジェクトがまだ存在することをチェックします。	クエリビジネスセキュリティプロファイルの作成: ビジネスレイヤにはオブジェクトがすでに存在しません。	警告

## 9.5.6 OLAP ユニバースのルールの整合性のチェック

整合性のチェックを使用して、OLAP ビジネスレイヤ内の次のルールをチェックできます。

表 9-8: すべてのオブジェクトに対する一般ルール

ルール名	説明	重要度
オブジェクト名のチェック	名前が一意で有効であることをチェックします。	エラー

表 9-9: 分析ディメンションルール

ルール名	説明	重要度
ネストされた分析ディメンションのチェック	分析ディメンションおよびそのフォルダに、別の分析ディメンションが含まれていないことをチェックします。	エラー

表 9-10: ディメンションルール

ルール名	説明	重要度
ディメンション内のメジャーのチェック	ディメンションにメジャーが含まれていないことをチェックします。	エラー
ネストされたディメンションのチェック	ディメンションに別のディメンションが含まれていないことをチェックします。	エラー
ディメンション内の分析のチェック	ディメンションに分析ディメンションが含まれていないことをチェックします。	エラー
属性参照のチェック	属性が同じ OLAP 特性を参照していることをチェックします。	エラー

表 9-11: 階層ルール

ルール名	説明	重要度
階層のチェック	階層に別の階層が含まれていないことをチェックします。	エラー
ディメンションのチェック	階層にディメンションが含まれていないことをチェックします。	エラー
分析ディメンションのチェック	階層に分析ディメンションが含まれていないことをチェックします。	エラー
階層内のメジャーのチェック	階層にメジャーが含まれていないことをチェックします。	エラー
属性のチェック	階層の属性が同じ OLAP 特性を参照していることをチェックします。	エラー
階層の MDX 式のチェック	階層の MDX 式を解析します。	エラー

表 9-12: 属性ルール

ルール名	説明	重要度
属性のチェック	属性に属性がないことをチェックします。	エラー
親ディメンションのチェック	属性に親ディメンションがあることをチェックします。	エラー
属性の MDX 式のチェック	属性の MDX 式を解析します。	エラー

表 9-13: メジャールール

ルール名	説明	重要度
集計関数の完全性のチェック	集計関数が空ではないことをチェックします。	エラー
メジャーのチェック	ビジネスレイヤに少なくとも1つのメジャーが含まれていることをチェックします。	警告
メジャーの MDX 式のチェック	メジャーの MDX 式を解析します。	エラー

表 9-14: フォルダルール

ルール名	説明	重要度
切断された依存のチェック	リンクされたオブジェクトを含むフォルダが失われていないことをチェックします。	エラー

表 9-15: ユニバースビュールール

ルール名	説明	重要度
ビューコンテンツのチェック	ビューが空ではないことをチェックします。	警告

### 9.5.7 OLAP ユニバース内の MDX オブジェクトのルールの整合性のチェック

次のルールでは、OLAP ユニバース内の MDX オブジェクトをチェックします。

表 9-16: オブジェクトメンバールール

ルール名	説明	重要度
必須のオブジェクト名	オブジェクト名が失われていないことをチェックします。	エラー
循環参照のチェック	式に @Select 関数が含まれている場合、循環参照がないことをチェックします。	エラー
@Prompt の解析の失敗	@Prompt 式が正しく解析することをチェックします。	エラー
@Variable の解析の失敗	@Variable 式が正しく解析することをチェックします。	エラー
MDX 式の確認	MDX 式が正しく解析することをチェックします。	エラー

表 9-17: 計算されたメジャー/メンバーのルール

ルール名	説明	重要度
解決順序	解決順序の値が数値であることをチェックします。	エラー
言語の値	言語の値が数値でないことをチェックします。	エラー

表 9-18: 計算されたメジャーのルール

ルール名	説明	重要度
計算されたメジャーの不正なタイプ	オブジェクトのデータタイプが式によって返されたデータタイプに一致することをチェックします。	エラー
計算されたメジャー必須の集計関数	集計関数が空ではないことをチェックします。	エラー
ビジネスレイヤで一意の計算されたメジャー名	名前が、ビジネスレイヤで定義済みの計算されたメジャーですすでに使用されていないことをチェックします。	エラー
計算されたメジャー名のサーバとの競合	名前が、OLAP サーバで定義済みのメジャーですすでに使用されていないことをチェックします(OLAP カタログで文字列 [Measures].[<計算されたメジャーの名前>] を検索)。	エラー

表 9-19: 計算メンバー固有のルール

ルール名	説明	重要度
計算メンバー必須の階層	階層が指定されていることをチェックします	重大
計算メンバー名のサーバとの競合	名前が、OLAP サーバで定義済みの名前付きセットで、すでに使用されていないことをチェックします。	警告

表 9-20: 特定の名前付きセットのルール

ルール名	説明	重要度
名前付きセット必須の階層	階層が指定されていることをチェックします	エラー
ビジネスレイヤで一意の名前付きセットの名前	名前が、ビジネスレイヤで定義済みの名前付きセットで、すでに使用されていないことをチェックします。	エラー
名前付きセット名のサーバとの競合	名前が、OLAP サーバで定義済みの名前付きセットで、すでに使用されていないことをチェックします。	警告
Crossjoin	式に文字列"crossjoin"が含まれていないことをチェックします。	警告





## リソースの公開

### 10.1 リソースの公開について

パブリケーションは、ユニバース作成プロセスの最後の手順です。ユニバースの公開ウィザードを使用して、ビジネスレイヤをローカルファイルシステムまたはリポジトリのいずれかに公開します。

ビジネスレイヤを公開すると、ウィザードはビジネスレイヤと、そらが参照するリソース(ローカル接続、接続ショートカット、データファンデーション) をエクスポートし、クエリ、レポート、分析ツールのユーザが利用できるユニバースを作成します。

ローカルで公開する。

ローカル接続で構築できるビジネスレイヤのみがローカルで公開できます。これには、ローカル OLAP 接続を基にしたビジネスレイヤ、またはローカル接続のある単一ソースのデータファンデーションを基にしたビジネスレイヤがあります。

公開されたユニバースは、指定したローカルファイルシステムフォルダに保存されます。

リポジトリへの公開

ユニバースをセキュリティ保護するには、Central Management Server (CMS) 上のリポジトリにまず公開する必要があります。ユニバースは、CMS のために定義されるオブジェクトレベルのセキュリティとユーザセキュリティ権限を継承します。ユニバースのデータとメタデータは、インフォメーションデザインツールのセキュリティエディタでセキュリティプロファイルを定義することにより、保護されます。

ローカルプロジェクトで接続を作成すると、それはセキュリティ保護されていないローカル接続であり、接続を参照するビジネスレイヤを公開する前に公開する必要があります。接続をセキュリティ保護するには、CMS のリポジトリに公開します。接続の公開ウィザードでは、セキュリティ保護された接続が作成され、ローカルプロジェクトの接続ショートカットが提供されます。

リポジトリに一旦公開されたリソースを参照し管理するには、リポジトリリソースビューを使用します。

公開されたリソースの編集

公開されたユニバースをインフォメーションデザインツールで直接編集することはできません。公開されたユニバースを操作するには、ユニバースの取得ウィザードを使用して取得する必要があります。ウィザードでは、ローカルフォルダまたはリポジトリからユニバースが取得され、ビジネスレイヤとそれが参照するリソース(ローカル接続、接続ショートカット、データファンデーション) に分割され、これらのリソースをユーザが編集できるローカルプロジェクトに作成します。

接続はリポジトリのみに公開されます。公開された接続を編集するには、リポジトリリソースビューから編集する必要があります。

## 関連項目

- ・ 170 ページの[ユニバースの公開](#)」
- ・ 35 ページの[公開されたユニバースの取得](#)」
- ・ 171 ページの[ローカル接続のリポジトリへの公開](#)」
- ・ 177 ページの[ユニバースセキュリティの概要](#)」
- ・ 173 ページの[リポジトリのリソースの管理について](#)」

## 10.2 ユニバースの公開

ユニバースをリポジトリに公開するには、ビジネスレイヤが1つまたは複数のセキュリティ接続ショートカットを参照している必要があります。すべてのショートカットは、ユニバースが公開されているリポジトリ内に定義された接続を、参照する必要があります。

**注**

ビジネスレイヤがローカル接続を参照していて、リポジトリに公開する場合は、まず接続を公開し、データファンデーション (リレーショナル) または接続ショートカットを使用するビジネスレイヤ (OLAP) で接続の参照を変更します。

ユニバースをローカルに公開するには、ビジネスレイヤが、リポジトリ内でセキュリティ保護されていないローカル接続のみを参照する必要があります。

## ユニバースを公開する前に推奨される操作

- ・ ビジネスレイヤおよびビジネスレイヤが参照するすべてのリソースを保存します。
  - ・ ビジネスレイヤが共有リソースを参照する場合は、公開されたユニバースにすべての変更が反映されるよう、プロジェクトを同期させます。
  - ・ ビジネスレイヤの整合性をチェックし、該当する場合は、データファンデーションの整合性もチェックします。ユニバース公開ウィザードでは、公開する前に整合性のチェックを実行するオプションがあります。
- 1 以下の手順を実行して、ユニバース公開ウィザードを起動します。
    - ・ リポジトリに公開するには、[ローカルプロジェクト]ビューでビジネスレイヤを選択し、ビジネスレイヤを右クリックして、[公開] > [リポジトリへ] を選択します。
    - ・ ローカルフォルダに公開するには、[ローカルプロジェクト]ビューでビジネスレイヤを選択し、ビジネスレイヤを右クリックして、[公開] > [ローカルフォルダへ] を選択します。
  - 2 ウィザードページの指示に従います。特定のページでの操作方法については、左下隅のヘルプアイコンをクリックしてください。

## 関連項目

- ・ 171 ページの[ローカル接続のリポジトリへの公開](#)」
- ・ 155 ページの[\[整合性のチェック\] について](#)」
- ・ 175 ページの[セッションを開く](#)」

## 10.2.1 リポジトリフォルダの選択

リポジトリにリソースを公開する、またはリポジトリからリソースを取得する場合、ウィザードの左側のペインにリポジトリ内のフォルダが表示されます。右側のペインにある表に、フォルダ内のリソースがリストされます。

リポジトリにリソースを公開する場合は、左側のペインのナビゲーションツリーでリポジトリフォルダに移動します。フォルダを挿入することができます。

公開されているユニバースを取得する場合、左側のペインでリポジトリフォルダに移動し、右側のペインでユニバースリストからユニバースを選択します。

### 注

デフォルトでは、リソースはローカルプロジェクト内に取得されます。取得されたデータファンデーションやビジネスレイヤを開くときに、ユーザは CMS 認証の入力を求められます。これにより、ローカルでのセキュリティが保護されます。

このローカルセキュリティ要件を削除するには、[すべてのユーザ用に保存] チェックボックスをオンにします。

## 10.2.2 ローカルフォルダの選択

ローカルフォルダ内のリソースを公開したり取得したりする場合、ウィザードでローカルフォルダのための指定を求められます。

- 1 ローカルマシンからアクセス可能なフォルダへのパスを入力します。
- 2 ファイルシステムを参照してフォルダを選択するには、[参照] をクリックします。

## 10.3 ローカル接続のリポジトリへの公開

- 1 [接続の公開] ウィザードを開始するには、[ローカルプロジェクト] ビューで接続を選択し、接続を右クリックして、[リポジトリへの接続の公開] を選択します。
- 2 ウィザードページの指示に従います。特定のページでの操作方法については、左下隅のヘルプアイコンをクリックしてください。

接続がリポジトリに公開されます。ローカル接続が [ローカルプロジェクト] ビューから削除されます。ローカルプロジェクト内で接続を作成するのに、選択肢があります。この接続に基づくビジネスレイヤを公開するには、新しいショートカットを参照するために、ビジネスレイヤまたはデータファンデーションを編集します。

関連項目

- ・ 175 ページの[セッションを開く](#)」
- ・ 51 ページの[接続ショートカットについて](#)」
- ・ 77 ページの[データファンデーション内での接続の変更](#)」
- ・ 107 ページの[ビジネスレイヤのデータソースの変更](#)」
- ・ 31 ページの[プロジェクトの同期](#)」

## リポジトリリソースの使用

### 11.1 リポジトリのリソースの管理について

リポジトリのリソースは、Central Management Server (CMS) 上のリポジトリでセキュリティ保護されているユニバースと接続です。リポジトリリソースビューでは、リポジトリ内のフォルダやリソース内をナビゲートしたりこれらと通信できます。[Connections] フォルダには、インフォメーションデザインツールとユニバースデザインツールを使って作成されるセキュリティ接続が含まれます。[Universe] フォルダには、インフォメーションデザインツール (.unx ユニバース) を使って公開されたユニバース、およびユニバースデザインツールを使って作成およびエクスポートされた、または旧バージョン (.unv ユニバース) から移行されたユニバースが含まれます。

リポジトリをナビゲートするには、リポジトリが保存されている CMS でセッションを開きます。セッションの詳細については、関連項目を参照してください。

以下の節では、リポジトリリソースビューから操作できるタスクの概要について説明します。

#### フォルダの管理

適切な権限があれば、[Connections] および [Universe] フォルダのサブフォルダを挿入、名前変更、削除できます。

#### セキュリティ接続の管理

- ・ 既存接続の編集
- ・ リポジトリに新しいセキュリティリレーショナルまたは OLAP 接続を挿入します。
- ・ 既存のセキュリティ接続からローカルプロジェクトに接続ショートカットを作成します。
- ・ リポジトリからセキュリティ接続を削除します。

#### ユニバース管理

インフォメーションデザインツールを使って公開された .unx ユニバースに以下のタスクを実行できます。

- ・ 整合性チェックを実行する。
- ・ クエリを実行する。このコマンドはクエリパネルを開きます。ユニバースのセキュリティプロファイルに定義されているセキュリティ設定は、セッションのユーザ名に従って適用されます。
- ・ ユニバースを取得する。このコマンドは、ビジネスレイヤとその参照されたリソースをユーザが編集できるようにローカルプロジェクトに保存します。
- ・ ユニバースの名前を変更する。このコマンドは、ユニバースの名前のみを変更し、基になるビジネスレイヤの名前は変更しません。
- ・ リポジトリからユニバースを削除する。

ユニバースデザインツールを使って作成した、または旧バージョンから移行した .unv ユニバースに以下のタスクを実行できます。

- ・ ユニバースを変換する。変換されたリソースをローカルプロジェクトに保存する、または変換された .unx ユニバースをリポジトリに公開します。
- ・ リポジトリからユニバースを削除する。

#### 関連項目

- ・ 174 ページの[セッション管理について](#)
- ・ 67 ページの[接続および接続ショートカットの編集について](#)
- ・ 53 ページの[リレーショナル接続の作成](#)
- ・ 62 ページの[OLAP 接続の作成](#)
- ・ 51 ページの[接続ショートカットについて](#)
- ・ 155 ページの[\[整合性のチェック\] について](#)
- ・ 176 ページの[リポジトリ内で公開されたユニバースでのクエリの実行](#)
- ・ 35 ページの[公開されたユニバースの取得](#)
- ・ 37 ページの[.unv ユニバースの変換について](#)

## 11.2 セッション管理について

セッションには、リポジトリに保存されているリソースにアクセスするために必要な Central Management Server (CMS) のシステム名と認証情報が含まれています。

セキュリティ保護されたリソースにアクセスする必要があるインフォメーションデザインツールのワークフローでは、[セッションを開く] ダイアログボックスが表示されます。アクセスするリポジトリのセッションをまだ定義していない場合は、[セッション] リストボックスから [新しいセッション] を選択します。セッションは、[リポジトリリソース] ビューの [セッションの挿入] コマンドで定義することもできます。

セッションを定義すると、[リポジトリリソース] ビューと [セッション] リストに保存されます。セッションを次回開いたときには、パスワードのみ入力する必要があります。

セッションを一旦開くと、インフォメーションデザインツールを終了するまで開いたままになります。セッションを明示的に閉じるには、[リポジトリリソース] ビューから閉じる必要があります。

セッションが異なる CMS システムにある限り、複数のセッションを同時に開くことができます。他にセッションが開いている CMS で異なるユーザ名およびパスワードを使ってセッションを開く必要がある場合は、開いているセッションをまず閉じる必要があります。

セッションが必要なくなってリストから削除するには、[リポジトリリソース] ビューの [セッションの削除] コマンドを使用します。

#### 関連項目

- ・ 179 ページの[インフォメーションデザインツール ユーザの CMC 権限](#)
- ・ 175 ページの[セッションを開く](#)
- ・ 176 ページの[セッションの終了](#)

### 11.2.1 セッションを開く

異なるワークフローでは、新たにセッションを開く必要があります。セッションを開くよう求められた場合、定義済みセッションの一覧が使用できます。[セッション] リストボックスは、次の順番で編成されています。

- ・ 開いているセッション (アルファベット順)
- ・ 閉じているセッション (アルファベット順)
- ・ 新しいセッション

インフォメーションデザインツールですでに定義済みのリポジトリでセッションを開くには、


- 1 次のいずれかを実行します。
  - ・ [リポジトリリソース] ビューで、リポジトリ名を右クリックし、[セッションを開く] を選択します。
  - ・ [セッション] リストボックスで、セッションを選択します。
- 2 ユーザ向けに CMS の認証情報が入力されます。セッションがまだ開いていない場合、[パスワード] を入力します。

#### 注

別のセッションがすでに開いているリポジトリでセッションを開こうとすると、エラーメッセージが表示されます。リポジトリでセッションを変更するには、まず、開いているセッションを [リポジトリリソース] ビューで閉じる必要があります。

- 3 ワークフローによって、[OK]、[次へ] または [接続] をクリックします。

インフォメーションデザインツールでまだ定義済みでないリポジトリでセッションを開くには、

- 1 次のいずれかを実行します。
  - ・ [リポジトリリソース] ビューで、[挿入]  メニューから、[セッションの挿入] を選択します。
  - ・ [セッション] リストボックスから [新しいセッション] を選択します。
- 2 [システム] ボックスで、リポジトリの保存場所の Central Management Server (CMS) 名を入力します。

#### 注

セッションを挿入するリポジトリが、このアプリケーションをホストしているクライアントとは別のドメインにあるマシンにホストされている場合は、クライアントの hosts ファイルにそのホスト情報を定義する必要があります。以下の場所にある hosts ファイルを書き換えます。

C:\WINDOWS\system32\drivers\etc\hosts

- 3 ユーザ名とパスワードを入力します。

#### 注

[認証] の種類が [Windows AD] の場合、[ユーザ名] に完全なドメイン名を指定します。たとえば、「myuser@domain」ではなく「myuser@domain.com」と入力します。

- 4 [認証] リストで、使用する認証方法を選択します。
- 5 ワークフローによって、[OK]、[次へ] または [接続] をクリックします。

セッションは、[リポジトリリソース] ビューで明示的にそのセッションを閉じるか、インフォメーションデザインツールを終了するまで開いたままになります。

#### 関連項目

- ・ 176 ページの[セッションの終了](#)

### 11.2.2 セッションの終了

インフォメーションデザインツールを終了すると、すべての開いているセッションが終了します。明示的にセッションを終了するには、次の手順に従います。

- 1 [リポジトリリソース] ビューで、終了するセッションを選択します。
- 2 右クリックし、[セッションを閉じる] を選択します。

### 11.3 リポジトリ内で公開されたユニバースでのクエリの実行

リポジトリ内で公開されたユニバースでクエリを実行する場合、クエリパネルは、セッションで定義されたユーザ名に従って、ユニバースのセキュリティプロファイル内に定義されている設定を適用します。

- 1 [リポジトリリソース] ビューで、ユニバースを選択します。

#### 注

.unx ユニバースのみを選択します。

- 2 ユニバース名を右クリックして、[クエリの実行] を選択します。

ユーザ名に対して許可されているビューとオブジェクトの一覧と一緒にクエリパネルが開きます。



## セキュリティの管理

### 12.1 インフォメーションデザインツールのセキュリティの概要

#### 12.1.1 ユニバースセキュリティの概要

ユニバースセキュリティは、ユニバースが Central Management Server (CMS) 上のリポジトリに公開されたときに開始されます。公開済みのユニバースは、[Universes] フォルダに、セキュリティ設定された接続は [Connections] フォルダにそれぞれ保存されます。

セントラル管理コンソール (CMC) を使用してシステムリポジトリ内に定義されるユーザおよびグループに基づくユニバースを保護します。

第 1 レベルのセキュリティとして、CMC を使用して、リポジトリ内の特定のフォルダ、リソース、ユニバース、および接続へのアクセス権を、特定のユーザおよびグループに付与します。これらの権限の定義方法については、『SAP Business Objects Business Intelligence platform 管理者ガイド』で説明しています。

デザイナーは、インフォメーションデザインツールのセキュリティエディタを使用して、別のセキュリティレベルを定義します。クエリ制限とコントロール、フィルタ、および行制限を使用するクエリで返されるデータを制限することもできます。ビジネスレイヤ内のオブジェクトおよびビューへのアクセスを許可、または拒否することもできます。このレベルのセキュリティを作成するには、ユニバースのセキュリティプロファイルを定義して、そのプロファイルをユーザやグループに割り当てます。この節では、セキュリティプロファイルの仕組みの基本について説明します。

##### セキュリティプロファイル

セキュリティプロファイルは、リポジトリで公開されたユニバースに適用される、セキュリティ設定グループです。この設定は、表示されるデータを制御し、データファンデーションおよびビジネスレイヤで定義されたパラメータを変更します。プロファイルがユーザまたはグループに割り当てられると、プロファイル内の設定によって、ユニバースに接続したときに表示されるオブジェクト、データおよび接続が決まります。プロファイルには、次の 2 種類があります。

- ・ データセキュリティプロファイルには、データファンデーション内およびデータ接続上のオブジェクトで定義されたセキュリティ設定があります。
- ・ ビジネスセキュリティプロファイルには、ビジネスレイヤ内のオブジェクトで定義されたセキュリティ設定があります。

各ユニバースに対して、複数のプロフィールを定義できます。プロフィールはリポジトリに保存されます。

### プロフィールの仕組み

CMC を使用してユニバースにアクセスする権限が許可されていて、セキュリティプロフィールが割り当てられていないか、セキュリティプロフィールを継承していないクエリツールおよびレポートツールユーザは、ユニバース内のすべてのオブジェクトおよび、これらのオブジェクトによって返されたすべてのデータを見ることができます。

プロフィールがユーザに割り当てられると、ユーザがユニバースでクエリを実行するたびに、プロフィール内で定義されているセキュリティ設定が適用されます。

インフォメーションデザインツールでは、リポジトリソースビューまたはセキュリティエディタからクエリを実行するときに、セキュリティプロフィールが適用されます。これらのプロフィールは、CMS セッションを開くために使用したユーザ名に従って適用されます。ビジネスレイヤエディタからクエリを実行する場合、セキュリティプロフィール設定は適用されません。

### 複数のプロフィールの扱い方

複数のプロフィールをユーザまたはグループに割り当てることができます。ユーザは、プロフィールを割り当てられている可能性があり、グループからプロフィールを継承する場合があります。ユーザに複数のプロフィールが割り当てられている場合、プロフィールはネットプロフィールという設定グループを作成するために集計されます。

集計は、優先順位と制限レベルに従います。これらはセキュリティエディタで変更できます。ユーザまたはグループが継承するプロフィールを表示して、ユーザまたはグループのネットプロフィールをプレビューできます。

### プロフィールのメンテナンス

プロフィールは、ユニバースそのものとは別に保存されます。ユニバースのデータファンデーションまたはビジネスレイヤ内に変更があっても、ユニバースの再公開時にプロフィールは影響を受けません。同様に、プロフィールの変更は割り当てとは独立しています。そのため、プロフィールの変更時に再割り当てをする必要はありません。すべての変更を含んだ上で、プロフィールは割り当てられたままです。

ユニバースを公開する場合、ユニバースで整合性のチェックを実行し、ユニバースおよびそのセキュリティプロフィール間に不一致があればフラグを付けます。

ユニバースに対して作成されたプロフィールは、そのユニバースが削除されるときに、削除されます。

### 関連項目

- ・ 182 ページの[データセキュリティプロフィール](#)」
- ・ 186 ページの[ビジネスセキュリティプロフィール](#)」
- ・ 191 ページの[セキュリティプロフィールの集計](#)」
- ・ 202 ページの[ユーザに割り当てられているプロフィールの表示とネットプロフィールのプレビュー](#)」
- ・ 176 ページの[リポジトリ内で公開されたユニバースでのクエリの実行](#)」
- ・ 199 ページの[セキュリティエディタの使用](#)」

### 12.1.2 インフォメーションデザインツールでのリソースのセキュリティについて

インフォメーションデザインツールの起動には認証は必要ありません。

ユーザは、[ローカルプロジェクト]ビューでデータファンデーション、ビジネスレイヤ、接続などのセキュリティ保護されていないリソースを作成および編集できます。リソースはローカルプロジェクトに保存されます。

リソースは、ユーザがローカルプロジェクトとそのリソースを共有するとき、またはユニバースまたは接続をリポジトリに公開するときにセキュリティ保護されます。共有されたプロジェクトと公開されたリソースは、Central Management Server (CMS) のリポジトリに安全に保存されます。

アプリケーション権限はセントラル管理コンソール (CMC) で付与されます。これらの権限の定義方法については、『SAP Business Objects Business Intelligence platform 管理者ガイド』で説明しています。

適切な権限を持つユーザは、公開されたユニバースをリポジトリから取得して編集することができます。また、リソースはプロジェクトの同期中に共有プロジェクトから取得することもできます。どちらの場合も、リソースはローカルプロジェクトの中に取得され、取得されたデータファンデーションやビジネスレイヤを開くときに、ユーザに CMS 認証の入力を求めることで、ローカルでセキュリティ保護されます。

インフォメーションデザインツールを起動すると、最後にツールを閉じたときに開いていたリソースが再度開かれます。セキュリティ保護されたリソースが開いているときは、ツールを起動するために CMS 認証情報を入力する必要があります。

#### 注

ローカルセキュリティ要件を削除するには、[すべてのユーザ用に保存] 権限が CMC で付与されている必要があります。リソースがすべてのユーザ用に保存されている場合は、ユーザは CMS 認証情報を入力しなくてもリソースを開けます。

セキュリティ接続をリポジトリから取得して、インフォメーションデザインツールでローカルに保存することはできません。その代わりに、リポジトリ内でのその接続へのショートカットがローカルプロジェクトに保存されます。セキュリティ接続は、リポジトリリソースビューからリポジトリで直接編集する必要があります。セキュリティ接続 (テーブル値の表示またはクエリの実行) からデータにアクセスするためには、ユーザは、接続が公開されたリポジトリの CMS 認証情報を入力する必要があります。システムは、認証情報を使用して、その接続に対しユーザが持つ権限を決定します。

#### 関連項目

- ・ 23 ページの [ローカルプロジェクトについて](#)
- ・ 51 ページの [接続ショートカットについて](#)

### 12.1.3 インフォメーションデザインツール ユーザの CMC 権限

インフォメーションデザインツールでタスクを実行するために必要なアプリケーション権限、ユニバース権限、および接続権限について、この項目で概説します。

権限はセントラル管理コンソール (CMC) で付与されます。これらの権限の定義方法については、『SAP Business Objects Business Intelligence platform 管理者ガイド』で説明しています。

リポジトリシステムでセッションを開き、インフォメーションデザインツール内ですべての保護されたタスクを実行するには、以下が必要です。

- ・ リポジトリが保存されている CMS 用に、CMS でシステム管理者によって設定されたユーザ名およびパスワードを持っている必要があります。
- ・ CMC で許可された インフォメーションデザインツールでの CMS への接続および CMC 内でのこのオブジェクトの表示権限を持っている必要があります。

必要なその他の権限は、タスク順に表に表示されています。

タスク	必要な権限
リポジトリに対する接続の公開	<ul style="list-style-type: none"> <li>・ [接続の作成、変更、または削除] アプリケーション権限</li> <li>・ 接続フォルダで [オブジェクトを表示する] 権限</li> <li>・ 接続フォルダで [オブジェクトをフォルダに追加する] 権限</li> </ul>
リポジトリリソースビューからセキュリティ接続を編集します。	<ul style="list-style-type: none"> <li>・ [接続の作成、変更、または削除] アプリケーション権限</li> <li>・ 接続フォルダでの [オブジェクトをフォルダに追加する] (作成用)</li> <li>・ [オブジェクトを編集する] 接続権限</li> </ul>
リポジトリに対するユニバースの公開	<ul style="list-style-type: none"> <li>・ [ユニバースの公開] アプリケーション権限</li> <li>・ ユニバースフォルダに対する [オブジェクトを表示する] 権限</li> <li>・ ユニバースフォルダに対する [オブジェクトをフォルダに追加する] 権限</li> <li>・ [オブジェクトを編集する] ユニバース権限 (再公開用)</li> </ul>
リポジトリからの公開済みユニバースの取得	<ul style="list-style-type: none"> <li>・ [ユニバースの取得] アプリケーション権限</li> <li>・ ユニバースフォルダに対する [オブジェクトを表示する] 権限</li> <li>・ [オブジェクトを表示する] ユニバース権限</li> <li>・ [ユニバースの取得] ユニバース権限</li> </ul>
保護されたローカルリソースの編集	<ul style="list-style-type: none"> <li>・ 権限は必要ありませんが、ユーザは、リソースを保存したユーザの CMS 認証を指定する必要があります。</li> </ul>

タスク	必要な権限
保護されていないローカルリソース	<ul style="list-style-type: none"> <li>・ [すべてのユーザ用に保存] アプリケーション権限</li> <li>・ [ユニバースの取得] アプリケーション権限</li> <li>・ ユニバースフォルダに対する [オブジェクトを表示する] 権限</li> <li>・ [オブジェクトを表示する] ユニバース権限</li> <li>・ [ユニバースの取得] ユニバース権限</li> <li>・ [すべてのユーザ用に保存] ユニバース権限</li> </ul>
セキュリティエディタを開く	<ul style="list-style-type: none"> <li>・ [セキュリティプロファイルの管理] アプリケーション権限</li> </ul>
セキュリティプロファイルの定義	<ul style="list-style-type: none"> <li>・ [オブジェクトを表示する] ユニバース権限</li> <li>・ [セキュリティプロファイルの編集] ユニバース権限</li> </ul>
ユーザおよびグループへのセキュリティプロファイルの割り当て	<ul style="list-style-type: none"> <li>・ [オブジェクトを表示する] ユニバース権限</li> <li>・ [セキュリティプロファイルの割り当て] ユニバース権限</li> </ul>
公開済みユニバースでのクエリの実行	<ul style="list-style-type: none"> <li>・ [オブジェクトを表示する] ユニバース権限</li> <li>・ [ユニバースに基づくクエリの作成と編集] ユニバース権限</li> <li>・ [データアクセス] ユニバース権限</li> <li>・ 基になる接続での [オブジェクトを表示する] 権限</li> <li>・ 基になる接続での [データアクセス] 権限</li> </ul>
プロジェクトリソースの共有は次のとおりです。 <ul style="list-style-type: none"> <li>・ ローカルプロジェクトの共有</li> <li>・ プロジェクトの同期ビューを開く</li> <li>・ プロジェクトリソースの同期</li> <li>・ リソースのロックおよびロック解除</li> <li>・ 共有プロジェクトの名前の変更または削除</li> </ul>	<ul style="list-style-type: none"> <li>・ [プロジェクトの共有] アプリケーション権限</li> </ul>
リポジトリに格納される.unv ユニバースの変換	<ul style="list-style-type: none"> <li>・ ユニバースフォルダに対する [オブジェクトを表示する] 権限</li> <li>・ ユニバースフォルダに対する [オブジェクトをフォルダに追加する] 権限</li> <li>・ [オブジェクトを表示する] ユニバース権限</li> </ul>
複数ソースユニバースの統計の計算	<ul style="list-style-type: none"> <li>・ [統計の計算] アプリケーション権限</li> <li>・ [オブジェクトを表示する] ユニバース権限</li> </ul>

タスク	必要な権限
ユニバースのリポジトリからの削除	<ul style="list-style-type: none"> <li>・ [オブジェクトを表示する] ユニバース権限</li> <li>・ [オブジェクトを削除する] ユニバース権限</li> </ul>
接続のリポジトリからの削除	<ul style="list-style-type: none"> <li>・ [接続の作成、変更、または削除] アプリケーション権限</li> <li>・ [オブジェクトを表示する] 接続権限</li> <li>・ [オブジェクトを削除する] 接続権限</li> </ul>

#### 関連項目

- ・ 174 ページの[セッション管理について](#)

## 12.2 データセキュリティプロファイル

データセキュリティプロファイルとは、データファンデーションおよびデータ接続内のオブジェクトを使用する公開済みユニバース上で、セキュリティを定義する設定のグループのことです。

すべてのデータセキュリティプロファイルの設定は、リレーショナルユニバースにのみ適用されます。

表 12-2: データセキュリティプロファイルのセキュリティ設定

セキュリティ設定	説明
接続	置換接続を定義します。
コントロール	置換クエリのタイムアウトとサイズの制限を定義します。
SQL	置換クエリのオプションを定義します。
行	クエリから返される行数を制限するために、SQL の WHERE 句を定義します。
テーブル	置換テーブルを定義します。

データセキュリティプロファイル設定の各種類については、関連項目で説明します。

#### 関連項目

- ・ 183 ページの[データセキュリティプロファイルの接続設定](#)
- ・ 183 ページの[データセキュリティプロファイルの制御設定](#)
- ・ 184 ページの[データセキュリティプロファイルの SQL 設定](#)
- ・ 185 ページの[データセキュリティプロファイルの行設定](#)
- ・ 186 ページの[データセキュリティプロファイルのテーブル設定](#)

- ・ 191 ページの[セキュリティプロファイルの集計](#)
- ・ 200 ページの[セキュリティプロファイルの挿入と編集](#)

## 12.2.1 データセキュリティプロファイルの接続設定

接続設定はリレーショナルユニバースでのみ定義できます。

データセキュリティプロファイルの接続設定を使用して、ユニバースで定義されている接続を上書きできる置換接続を定義します。置換接続を含むプロファイルが割り当てられるかプロファイルを継承すると、ユニバースでクエリを実行するときに、ユニバースで定義された接続の代わりに、置換接続が使用されます。

セキュリティ接続のみを置換接続として定義できます。リレーショナル接続は、次の3つのタイプのいずれかになります。置換接続は、元の接続と同じタイプである必要があります。

- ・ SAP NetWeaver BW リレーショナルデータベース
- ・ SAP リレーショナルデータベース
- ・ 別のリレーショナルデータベース

セキュリティプロファイルを定義しているリポジトリで許可された、“オブジェクトを表示する” 権限を持つ接続フォルダおよびサブフォルダで、接続を選択できます。

複数の接続を利用する複数ソースユニバースの場合、各接続に対して置換を定義できます。

### 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)

## 12.2.2 データセキュリティプロファイルの制御設定

制御設定はリレーショナルユニバースでのみ定義できます。

データセキュリティプロファイルの制御設定を使用して、データベースからデータを取得するときにデフォルトの制限を上書きする置換クエリの制限を定義します。デフォルトのクエリ制限は、ユニバース作成者がビジネスレイヤで設定します。置換制御設定を使用してプロファイルが割り当てられるかプロファイルを継承すると、クエリを実行するときに、ビジネスレイヤプロパティで定義された制限の代わりに、置換制限が使用されます。

データセキュリティプロファイル用のエディタに、選択された制限とビジネスレイヤで定義された制限値が表示されます。制限を選択または選択解除するか、制限の新しい値を入力すると、ラベルの表示が太字に変更されます。これは、この制限が上書きされたもので、ユニバースで定義されたデフォルト制限ではないことを示します。

クエリ制限	入力される値
結果の最大行数	<ul style="list-style-type: none"><li>TRUE および 0 ～ 2147483647 行の間の数値サイズ</li><li>False</li></ul>
最大実行時間	<ul style="list-style-type: none"><li>TRUE および 0 ～ 2147483647 分の間の数値サイズ</li><li>FALSE</li></ul>
予測所要時間が超えたら警告	<ul style="list-style-type: none"><li>TRUE および 0 ～ 10000 分の間の数値サイズ</li><li>False</li></ul>

クエリ制限に関する詳細については、ビジネスレイヤのプロパティについての関連項目を参照してください。

#### 関連項目

- ・ 104 ページの[ビジネスレイヤのプロパティについて](#)
- ・ 191 ページの[セキュリティプロファイルの集計](#)

### 12.2.3 データセキュリティプロファイルの SQL 設定

SQL 設定はリレーショナルユニバースでのみ定義できます。

データセキュリティプロファイルの SQL 設定を使用して、置換クエリオプションを定義します。ユニバース作成者は、ビジネスレイヤのデフォルトの各種クエリオプションとデータファンデーションの各種プロパティを定義します。SQL 設定を使用してプロファイルが割り当てられるかプロファイルを継承すると、クエリパネルを使用するときに、ユニバースで定義されたクエリオプションの代わりに、置換オプションが使用されます。

データセキュリティプロファイル用のエディタに、ビジネスレイヤで選択された SQL 設定およびデータファンデーションが表示されます。オプションを選択または選択解除すると、ラベルの表示が太字に変更されます。これは、このオプションが上書きされたもので、ユニバースに定義されたデフォルトのものではないことを示します。



クエリオプション	入力される値
サブクエリの使用	<ul style="list-style-type: none"> <li>TRUE</li> <li>FALSE</li> </ul>
ユニオン、インターセクト、マイナス演算子の使用	<ul style="list-style-type: none"> <li>TRUE</li> <li>FALSE</li> </ul>
クエリパネルでの複合演算子の使用	<ul style="list-style-type: none"> <li>TRUE</li> <li>FALSE</li> </ul>
各コンテキストに対して複数の SQL 文を生成	<ul style="list-style-type: none"> <li>TRUE</li> <li>FALSE</li> </ul>
各メジャーに対して複数の SQL 文を生成	<ul style="list-style-type: none"> <li>TRUE</li> <li>FALSE</li> </ul>
デカルト積を許可する	<ul style="list-style-type: none"> <li>警告</li> <li>使用不可</li> </ul>

クエリオプションに関する詳細については、ビジネスレイヤおよびデータファンデーションのプロパティについての関連項目を参照してください。

#### 関連項目

- 104 ページの[ビジネスレイヤのプロパティについて](#)
- 93 ページの[データファンデーションのプロパティについて](#)
- 191 ページの[セキュリティプロファイルの集計](#)

## 12.2.4 データセキュリティプロファイルの行設定

行設定はリレーショナルユニバースでのみ定義できます。

データセキュリティプロファイルの行設定を使用して、クエリから返される行数を制限します。特定のテーブルの SQL の WHERE 句を定義して、行数を制限します。行設定を使用してプロファイルが割り当てられるかプロファイルを継承すると、ユニバースでクエリを実行するときに、テーブルがクエリで参照されている場合に生成される SQL に、定義された WHERE 句が追加されます。

#### 注

生成された SQL をレポートツールで編集する権限のあるユーザは、行設定で生成された WHERE 句を変更できます。ユーザが SQL を変更しようと、レポートツールでユーザの権限を管理してください。

データファンデーション内のいずれの標準テーブルでも WHERE 句を定義できます。WHERE 句の SQL には次のものが含まれます。

- ・ @Variable および @Prompt などの @関数
- ・ 複数ソースが有効なユニバースの場合、ユニバースで定義されたすべての接続内にある別のテーブルへの参照
- ・ 複数ソース有効ユニバースの場合、SAP BusinessObjects SQL 関数

WHERE 句の SQL に次のものは含まれません。

- ・ 計算列
- ・ 派生テーブル

#### 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)

## 12.2.5 データセキュリティプロファイルのテーブル設定

テーブル設定はリレーショナルユニバースでのみ定義できます。

データセキュリティプロファイルのテーブル設定を使用して、置換テーブルを定義します。テーブル設定を使用してプロファイルが割り当てられるかプロファイルを継承すると、元のテーブルを参照するクエリを実行するときに、代わりに置換テーブルが使用されます。

#### 注

生成された SQL をレポートツールで編集する権限のあるユーザは、置換テーブル名を変更できます。ユーザが SQL を変更しように、レポートツールでユーザの権限を管理してください。

データファンデーションの標準テーブルを、ユニバースに定義されたいずれかの接続のデータベーステーブルか、データファンデーション内の他の標準テーブルに置き換えることができます。

#### 注

置換テーブルの所有者と修飾子を指定する場合は、テーブル名の一部としてではなく、提供されるフィールドにこれらを入力する必要があります。データファンデーションのテーブル名の詳細については、関連項目を参照してください。

#### 関連項目

- ・ 78 ページの[データファンデーションのテーブルについて](#)
- ・ 191 ページの[セキュリティプロファイルの集計](#)

## 12.3 ビジネスセキュリティプロファイル

ビジネスセキュリティプロファイルとは、ビジネスレイヤ内にあるオブジェクトを使用する公開済みユニバースで、セキュリティを定義する設定グループです。

表 12-3: ビジネスセキュリティプロファイルのセキュリティ設定

セキュリティ設定	説明
クエリの作成	クエリパネルでユーザが使用できるユニバースビューおよびビジネスレイヤオブジェクトを定義します。  <b>注</b> クエリの作成設定では、メタデータのみを保護します。
表示データ	ユーザがクエリを実行する際に、ビジネスレイヤ内のオブジェクトによって取得されるデータへのアクセスを許可または拒否します。
フィルタ	ビジネスレイヤ内のオブジェクトを使用するフィルタを定義します。

ビジネスセキュリティプロファイル設定の各種類については、関連項目で説明します。

ビジネスレイヤでは、デザイナーがオブジェクトのステータスを[有効]、[非表示]、または[使用停止]に設定できます。プロファイル設定の定義中は、ビジネスレイヤ内のすべての有効なオブジェクトにアクセスできます。ビジネスレイヤで非表示または使用停止に設定されたオブジェクトは、クエリパネルまたはレポートには一切表示されません。

#### 関連項目

- ・ 187 ページの[ビジネスセキュリティプロファイルのクエリの作成設定](#)
- ・ 189 ページの[ビジネスセキュリティプロファイルの表示データ設定](#)
- ・ 190 ページの[ビジネスセキュリティプロファイルのフィルタ設定](#)
- ・ 191 ページの[セキュリティプロファイルの集計](#)
- ・ 200 ページの[セキュリティプロファイルの挿入と編集](#)

### 12.3.1 ビジネスセキュリティプロファイルのクエリの作成設定

ビジネスセキュリティプロファイルのクエリの作成設定を使用して、クエリパネルにおけるビジネスレイヤオブジェクトの使用を許可または拒否します。

デフォルトでは、リポジトリで許可されたユニバースへのアクセス権を持つユーザは、クエリパネルですべてのユニバースオブジェクトを見ることができます。クエリの作成設定を持つプロファイルが割り当てられるかプロファイルを継承すると、その設定によって許可されたビューとオブジェクトのみが表示され、クエリで選択できるようになります。

明示的に許可も拒否もされていないオブジェクトは、デフォルトで拒否されます。明示的に拒否されたオブジェクトとは異なり、デフォルトで拒否されたオブジェクトは、ビジネスセキュリティプロファイルを集計してユーザの

ネットプロファイルを決定した後に継承することで、許可される可能性があります。プロファイルの集計の詳細については、関連項目を参照してください。

オブジェクトの許可および拒否には、次の 2 種類の方法があります。

- ・ ビジネスレイヤビューを使用する: ビューのすべてのオブジェクトを許可または拒否します。[すべてのビジネスレイヤビュー] オプションを使用すると、ユニバースに定義されたすべてのビューを許可または拒否できます。
- ・ オブジェクトを使用する: 次のオブジェクトを許可または拒否できます。[すべてのオブジェクト] オプションを使用すると、ビジネスレイヤのすべてのオブジェクトを許可または拒否できます。
  - ・ ディメンション
  - ・ 属性
  - ・ メジャー
  - ・ 計算メンバー
  - ・ フィルタ
  - ・ プロンプト
  - ・ 名前付きセット
  - ・ フォルダ: フォルダ内のすべてのオブジェクトを許可または拒否
  - ・ 分析ディメンション: ディメンション内のすべてのオブジェクトを許可または拒否
  - ・ 階層: 階層内のすべてのオブジェクトを許可または拒否

#### 注

階層レベルの許可または拒否はできません。

#### ヒント

ほとんどのビューが許可されている場合は、すべてのビューを許可してから、許可しないビューを拒否すると操作が簡単になります。[すべてのビジネスレイヤビュー] オプションおよび [すべてのオブジェクト] オプションを使用すると、ユニバースが公開されるときに、そのビジネスレイヤで定義された新しいビューやオブジェクトを自動的にクエリ作成設定に含めることができます。

[すべてのビジネスレイヤビュー] または [すべてのオブジェクト] オプションが使用されると、このプロファイルのネット設定を決定するために設定が集計されます。例:

- ・ [すべてのビジネスレイヤビュー] が拒否され、ビューが 1 つ許可されている場合、このプロファイルは許可されている 1 つ以外のすべてのビューを拒否します。
- ・ [すべてのビジネスレイヤビュー] が許可され、ビューが 1 つだけ拒否されている場合、このプロファイルは拒否されている 1 つ以外のすべてのビューを許可します。
- ・ [すべてのオブジェクト] が拒否され、オブジェクトが 1 つだけ許可されている場合、そのオブジェクトにアクセスするパスに限り、そのパス上のすべての親フォルダが許可されます。親フォルダ内の他のオブジェクトは拒否されます。
- ・ [すべてのオブジェクト] が許可され、オブジェクトが 1 つだけ拒否されている場合、そのオブジェクトにアクセスするパスに限り、そのパス上の親フォルダが拒否されます。親フォルダ内の他のオブジェクトは許可されます。

許可されたビューのオブジェクトは、そのビューでだけ許可されます。同じオブジェクトが他のビューに含まれる場合、自動的に許可されません。

あるユーザがクエリパネルで特定のオブジェクトを見られるかどうかは、そのユーザに割り当てられているすべてのプロファイル内のクエリの作成設定が集計され、オブジェクトアクセスレベルが考慮された後で決まります。プロファイルの集計の詳細については、関連項目を参照してください。

## 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)

## 12.3.2 ビジネスセキュリティプロファイルの表示データ設定

ビジネスセキュリティプロファイルの表示データ設定を使用して、ビジネスレイヤ内でオブジェクトが取得したデータへのアクセスを許可または拒否します。

デフォルトでは、リポジトリで許可されたユニバースへのアクセス権を持つユーザは、すべてのユニバースオブジェクトが取得したデータを見ることができます。表示データ設定を使用してプロファイルが割り当てられるかプロファイルを継承すると、その設定によって許可されたオブジェクトに対応するデータだけが表示されるようになります。

明示的に許可も拒否もされていないオブジェクトは、デフォルトで拒否されます。明示的に拒否されたオブジェクトとは異なり、デフォルトで拒否されたオブジェクトは、ビジネスセキュリティプロファイルを集計してユーザのネットプロファイルを決定的後に継承することで、許可される可能性があります。プロファイルの集計の詳細については、関連項目を参照してください。

許可または拒否できるのは次のオブジェクトです。[すべてのオブジェクト] オプションを使用すると、ビジネスレイヤのすべてのオブジェクトを許可または拒否できます。

- ・ ディメンション
- ・ 属性
- ・ メジャー
- ・ 計算メジャー
- ・ 計算メンバー
- ・ 名前付きセット
- ・ フォルダ: フォルダ内のすべてのオブジェクトを許可または拒否
- ・ 階層

[すべてのオブジェクト] オプションを使用すると、ユニバースが公開されるときに、そのビジネスレイヤで定義された新しいオブジェクトを自動的に表示データ設定に含めることができます。

[すべてのオブジェクト] オプションが使用されると、このプロファイルのネット設定を決定するために設定が集計されます。例:

- ・ [すべてのオブジェクト] が拒否され、オブジェクトが 1 つだけ許可されている場合、そのオブジェクトにアクセスするパスに限り、そのパス上のすべての親フォルダが許可されます。親フォルダ内の他のオブジェクトは拒否されます。
- ・ [すべてのオブジェクト] が許可され、オブジェクトが 1 つだけ拒否されている場合、そのオブジェクトにアクセスするパスに限り、そのパス上の親フォルダが拒否されます。親フォルダ内の他のオブジェクトは許可されます。

表示データ設定で特定のオブジェクトを拒否するように設定しているユーザが、拒否したオブジェクトを含むレポートを最新表示することがあります。ビジネスレイヤで SQL 生成パラメータである AUTO\_UPDATE\_QUERY を設定すると、この場合に最新表示が必要な内容を指定できます。

- ・ このパラメータが [いいえ] に設定されている場合、レポートの最新表示がエラーメッセージを生成します。

- ・ このパラメータが [はい] に設定されている場合、拒否されたオブジェクトはクエリとそのビジネスレイヤで定義されたすべてのフィルタから削除されます。その他の許可されたオブジェクトのデータは、取得されて部分的なレポート内でユーザに表示されます。

あるユーザが特定のオブジェクトのデータを見られるかどうかは、そのユーザに割り当てられているすべてのプロファイル内の表示データ設定が集計され、オブジェクトアクセスレベルが考慮された後で決まります。プロファイルの集計の詳細については、関連項目を参照してください。

#### 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)

### 12.3.3 ビジネスセキュリティプロファイルのフィルタ設定

ビジネスセキュリティプロファイルのフィルタ設定を使用して、ビジネスレイヤまたは指定されたメンバーセットで、オブジェクトを使用するフィルタを定義します。セキュリティエディタを使用して、ビジネスセキュリティプロファイルのフィルタを明示的に作成および編集します。ビジネスセキュリティプロファイルのフィルタは、ビジネスレイヤではアクセスできません。ビジネスセキュリティプロファイルが削除されると、フィルタまたは名前付きセットも削除されます。

フィルタ設定を使用してプロファイルが割り当てられるかプロファイルを継承すると、フィルタがクエリスクリプトに追加され(またそのために、ビジネスレイヤ内に定義されているすべてのフィルタに結合され)、表示されるデータが制限されます。

#### リレーショナルユニバース

リレーショナルユニバースの場合、ビジネスレイヤ内のディメンションおよびメジャーのフィルタを定義します。AND または OR 演算子にリンクされる複合フィルタを定義することができます。また、クエリに適用する複数のフィルタを定義することもできます。

ユーザがクエリを実行すると、このフィルタがクエリと返されるデータに常に適用されます。これは、定義されたテーブルがクエリで参照されている場合のみ適用される、データセキュリティプロファイルの行設定とは異なります。

#### OLAP ユニバース

OLAP ユニバースの場合、メンバーの名前付きセットを定義します。ビジネスレイヤで、すべてのディメンションのメンバーを含めるか除外することができます。除外メンバーは、データがキューブから取得される際、クエリから削除されます。

#### 注

このフィルタはレポート内の値の集計に影響を与えません。メンバーの表示がフィルタリングされるだけです。

複数のディメンションからメンバーを含めたり除外したりすることができます。また、クエリに適用する複数の名前付きセットを定義することもできます。

#### 関連項目

- ・ 141 ページの[メンバーセレクトについて](#)

- ・ 112 ページの[フィルタについて](#)
- ・ 191 ページの[セキュリティプロファイルの集計](#)

## 12.4 セキュリティプロファイルの集計

1 つのユニバースに定義された複数のデータセキュリティプロファイルまたはビジネスセキュリティプロファイルが、同じユーザに割り当てられている可能性があります。複数のプロファイルを、1 人のユーザまたは 1 つのグループに直接割り当てることができ、親グループから継承することもできます。この場合、異なるプロファイル内のセキュリティ設定は、ネットプロファイルという 1 つの実効データセキュリティプロファイル、および 1 つの実効ビジネスセキュリティプロファイルに集計されます。ネットプロファイル内の設定は、ユーザがクエリを作成するか、レポートを表示するときに適用されます。

セキュリティ設定の集計には 2 種類の方法が使用されます。優先順位と制限レベルです。

優先順位は、データセキュリティプロファイル設定の集計に使用されます。セキュリティエディタでデータセキュリティプロファイルの優先順位を指定できます。

データセキュリティプロファイル設定の一部とすべてのビジネスセキュリティプロファイル設定は、非常に厳しい制限、中程度の制限、緩い制限のいずれかの制限レベルに基づいて集計されます。

制限レベルは、プロファイルの集計に使用する演算子を定義します。プロファイルが継承されたのか結合されたのかによって、異なる集計演算子が使用されます。

- ・ ユーザまたはグループに、プロファイル A が割り当てられており、プロファイル B が割り当てられたグループに属している場合、プロファイル A およびプロファイル B が継承されます。
- ・ ユーザまたはグループが、プロファイル A が割り当てられたグループと、プロファイル B が割り当てられた別のグループに属している場合、プロファイル A およびプロファイル B が割り当てられます。
- ・ ユーザまたはグループに、プロファイル A とプロファイル B の両方が割り当てられている場合、プロファイル A およびプロファイル B が結合されます。

プロファイルの集計方法を指定する場合は、セキュリティエディタで制限レベルを変更することができます。

- ・ セキュリティがロールを考慮してデザインされており、各ロールがユーザに新しい権限を許可する場合、[緩い制限] レベルが最適です。
- ・ 各プロファイルがユーザに対する表示内容を制限するために使用されている場合、最も厳しい制限レベルが最適です。
- ・ [中程度の制限] レベルでは、継承されたプロファイルに対し最も厳しい制限レベルを使用し、結合されたプロファイルに対し緩い制限を使用します。

プロファイル設定を集計するのに使用される演算子 (AND、OR など) は、設定によって異なります。各設定の種類の集計に関する詳細情報は、関連項目を参照してください。

データセキュリティプロファイルの行設定と、ビジネスセキュリティプロファイルのフィルタ設定の両方で、クエリにフィルタを適用するために WHERE 句が生成されます。最初に行設定が適用されます。次に、フィルタ設定内の WHERE 句が最初のクエリ結果に適用されます。実際には、2 つの WHERE 句が AND 演算子で集計されます。



**関連項目**

- ・ 192 ページの[接続設定の集計](#)」
- ・ 192 ページの[制御設定の集計](#)」
- ・ 193 ページの[SQL 設定の集計](#)」
- ・ 194 ページの[行設定の集計](#)」
- ・ 195 ページの[テーブル設定の集計](#)」
- ・ 195 ページの[クエリの作成設定の集計](#)」
- ・ 197 ページの[表示データ設定の集計](#)」
- ・ 198 ページの[フィルタ設定の集計](#)」
- ・ 201 ページの[データセキュリティプロファイルの優先順位の変更](#)」
- ・ 201 ページの[セキュリティプロファイルの集計オプションの変更](#)」

### 12.4.1 接続設定の集計

ユニバースの 2 つ以上のデータセキュリティプロファイルが、同じユーザーに割り当てられているか、同じユーザーによって継承されている場合、データセキュリティプロファイル内で最も高い優先度で定義されている接続が使用されます。

ユニバースに複数の接続がある場合、接続設定の集計は接続ごとに行われます。

**関連項目**

- ・ 201 ページの[データセキュリティプロファイルの優先順位の変更](#)」
- ・ 183 ページの[データセキュリティプロファイルの接続設定](#)」

### 12.4.2 制御設定の集計

ユニバースの 2 つ以上のデータセキュリティプロファイルが、同じユーザーに割り当てられているか、同じユーザーによって継承されている場合、制御設定の集計には次のルールが使用されます。このルールは、ユーザーがクエリやレポートを実行するときに使用される値を決定するために、各クエリ制限に適用されます。



制限レベル	集計ルール
非常に厳しい制限	<p>この制限は、すべての結合されたプロファイルと継承されたプロファイルで選択されている場合のみ有効です。</p> <p>使用される値は、すべての結合されたプロファイルと継承されたプロファイルにおける、その制限の最小値です。</p>
中程度の制限	<p>この制限は、すべての継承されたプロファイルで選択されており、少なくとも 1 つの結合されたプロファイルで選択されている場合にのみ有効です。</p> <p>最初に、継承されたプロファイルを比較して、その制限の最小値が決定されます。次に、この値は結合されたプロファイル内の値と比較されます。これらの値のうち最大の値が使用されます。</p>
緩い制限	<p>この制限は、いずれかの結合されたプロファイルまたは継承されたプロファイルで選択されている場合に有効です。</p> <p>使用される値は、すべての結合されたプロファイルと継承されたプロファイルにおける、その制限の最大値です。</p>
優先順位 (デフォルト)	最も高い優先順位を持つデータセキュリティプロファイルの制限の有効化と値が使用されます。

**注**

継承したプロファイルおよび結合されたプロファイルの定義については、セキュリティプロファイルの集計に関する項目を参照してください。

**関連項目**

- ・ 191 ページの[セキュリティプロファイルの集計](#)
- ・ 201 ページの[セキュリティプロファイルの集計オプションの変更](#)
- ・ 201 ページの[データセキュリティプロファイルの優先順位の変更](#)
- ・ 183 ページの[データセキュリティプロファイルの制御設定](#)

**12.4.3 SQL 設定の集計**

ユニバースの 2 つ以上のデータセキュリティプロファイルが、同じユーザに割り当てられているか、同じユーザによって継承されている場合、SQL 設定の集計には次のルールが使用されます。このルールは、ユーザがクエリを作成するときに使用される値を決定するために、各クエリオプションに適用されます。

制限レベル	集計ルール
非常に厳しい制限	<p>このオプションは、すべての結合されたプロファイルと継承されたプロファイルで選択されている場合のみ有効です。</p> <p>デカルト積では、すべての結合されたプロファイルと継承されたプロファイルで、値が“使用不可”の場合のみ、“使用不可”が使用されます。</p>
中程度の制限	<p>このオプションは、すべての継承されたプロファイルで選択されており、少なくとも 1 つの割り当てられたプロファイルで選択されている場合に有効です。</p> <p>デカルト積では、すべての継承されたプロファイルで、値が“使用不可”であり、少なくとも 1 つの結合されたプロファイルで“使用不可”の場合、“使用不可”が使用されます。</p>
緩い制限	<p>このオプションは、いずれかの結合されたプロファイルまたは継承されたプロファイルで選択されている場合に有効です。</p> <p>デカルト積では、いずれかの結合されたプロファイルまたは継承されたプロファイルで、値が“警告”の場合、“警告”が使用されます。</p>
優先順位 (デフォルト)	最も高い優先順位を持つデータセキュリティプロファイルのオプションの有効化と値が使用されます。

**注**

継承したプロファイルおよび結合されたプロファイルの定義については、セキュリティプロファイルの集計に関する項目を参照してください。

**関連項目**

- ・ 191 ページの[セキュリティプロファイルの集計](#)
- ・ 201 ページの[セキュリティプロファイルの集計オプションの変更](#)
- ・ 201 ページの[データセキュリティプロファイルの優先順位の変更](#)
- ・ 184 ページの[データセキュリティプロファイルの SQL 設定](#)

**12.4.4 行設定の集計**

ユニバースの 2 つ以上のデータセキュリティプロファイルが、同じユーザに割り当てられているか、同じユーザによって継承されている場合、ユーザがクエリやレポートを実行するとき、行設定の集計と使用される WHERE 句の決定には、次のルールが使用されます。

最初に、各テーブルの WHERE 句が制限レベルに従って集計されます。

制限レベル	集計ルール
非常に厳しい制限(デフォルト)	同じテーブルに適用されるすべてのプロファイルの WHERE 句が、AND 演算子で結合されます。
中程度の制限	継承された WHERE 句が、AND 演算子を使用して集計されます。 結合された WHERE 句が、OR 演算子を使用して集計されます。
緩い制限	同じテーブルに適用されるすべてのプロファイルの WHERE 句が、OR 演算子で結合されます。

制限レベルに従った集計の後、各テーブルの WHERE 句がまとめて AND 演算子で集計され、クエリに適用される最終的な WHERE 句が作成されます。

#### 注

継承したプロファイルおよび結合されたプロファイルの定義については、セキュリティプロファイルの集計に関する項目を参照してください。

#### 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)
- ・ 201 ページの[セキュリティプロファイルの集計オプションの変更](#)
- ・ 185 ページの[データセキュリティプロファイルの行設定](#)

## 12.4.5 テーブル設定の集計

ユニバースの 2 つ以上のデータセキュリティプロファイルが、同じユーザに割り当てられているか、同じユーザによって継承されている場合、データセキュリティプロファイル内で最も高い優先度で定義されている置換テーブルが使用されます。複数のテーブルに設定が定義されている場合、集計はテーブルごとに行われます。

#### 関連項目

- ・ 201 ページの[データセキュリティプロファイルの優先順位の変更](#)
- ・ 186 ページの[データセキュリティプロファイルのテーブル設定](#)

## 12.4.6 クエリの作成設定の集計

ユニバースの 2 つ以上のビジネスセキュリティプロファイルが、同じユーザに割り当てられているか、同じユーザによって継承されている場合、クエリの作成設定が集計されます。オブジェクトアクセスレベル (定義されている場合) は、ユーザがクエリパネルで指定のオブジェクトを表示するかどうかを決定するために適用されます。

最初に、クエリパネルでユーザが選択できるビューの一覧が、制限レベルに従ってプロファイルを集計することで決定されます。

制限レベル	集計ルール
非常に厳しい制限(デフォルト)	ユーザは、ビューがすべての継承したプロファイルおよび結合されたプロファイルで許可されている場合のみ、クエリパネルでそのビューを選択できます。
中程度の制限	ユーザは、ビューがすべての継承したプロファイルで許可されており、また、少なくとも 1 つの結合されたプロファイルで許可されている場合のみ、クエリパネルでそのビューを選択できます。
緩い制限	ユーザは、ビューが継承したプロファイルまたは結合されたプロファイルのいずれかで許可されている場合、クエリパネルでそのビューを選択できます。

クエリパネルでビューが選択されると、オブジェクトがビューに含まれている場合、また、次の制限レベルに従ってプロファイルを集計した後で明示的に拒否されていない場合、そのオブジェクトが表示されます。

制限レベル	集計ルール
非常に厳しい制限(デフォルト)	すべての継承したプロファイルおよび結合されたプロファイルで明示的に拒否されている場合、オブジェクトは拒否されます。
中程度の制限	継承したプロファイルのいずれかと、すべての結合されたプロファイルで明示的に拒否されている場合、オブジェクトは拒否されます。
緩い制限	すべての継承したプロファイルおよび結合されたプロファイルで明示的に拒否されている場合のみ、オブジェクトは拒否されます。

集計後、拒否されたオブジェクトは、許可されたビューに含まれている場合でも、表示されません。フォルダが拒否されると、次に、そのフォルダ内のすべてのサブフォルダとオブジェクトが拒否されます。

最後に、セントラル管理コンソールでユーザに許可されたアクセスレベルに応じて、実効ビジネスセキュリティプロファイルから許可されるオブジェクトで、クエリパネルで使用できるものが決定されます。ユーザは、認証されたアクセスレベルより低いアクセスレベルか、同等のアクセスレベルを持つオブジェクトのみ表示できます。オブジェクトに対するアクセスレベルは、ビジネスレイヤエディタで割り当てます。

#### 注

継承したプロファイルおよび結合されたプロファイルの定義については、セキュリティプロファイルの集計に関する項目を参照してください。

オブジェクトのアクセスレベルについての詳細は、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。

## 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)
- ・ 201 ページの[セキュリティプロファイルの集計オプションの変更](#)
- ・ 187 ページの[ビジネスセキュリティプロファイルのクエリの作成設定](#)

## 12.4.7 表示データ設定の集計

ユニバースの 2 つ以上のビジネスセキュリティプロファイルが、同じユーザに割り当てられているか、同じユーザによって継承されている場合、表示データ設定が集計されます。オブジェクトアクセスレベル(定義されている場合)は、ユーザがビジネスレイヤでオブジェクトのデータを表示するかどうかを決定するために適用されます。

まず、制限レベルに従ってプロファイルを集計することにより、ユーザが表示できるオブジェクトの一覧が決定されます。

制限レベル	集計ルール
非常に厳しい制限(デフォルト)	すべての継承されたプロファイルと結合されたプロファイルで許可されている場合のみ、データが表示されます。
中程度の制限	すべての継承されたプロファイルと、少なくとも 1 つの結合されたプロファイルで、オブジェクトが許可されている場合のみ、データが表示されます。
緩い制限	いずれかの継承されたプロファイルまたは結合されたプロファイルで、オブジェクトが許可されている場合、データが表示されます。

フォルダが拒否されると、次に、フォルダ内のすべてのオブジェクトのデータと、そのサブフォルダが拒否されます。

最後に、セントラル管理コンソールでユーザに許可されたアクセスレベルに応じて、ユーザがデータを表示することを実効ビジネスセキュリティプロファイルから許可されるオブジェクトが決定されます。ユーザは、認証されたアクセスレベルより低いアクセスレベルか、同等のアクセスレベルを持つオブジェクトのデータのみ表示できます。オブジェクトに対するアクセスレベルは、ビジネスレイヤエディタで割り当てます。

**注**

継承したプロファイルおよび結合されたプロファイルの定義については、セキュリティプロファイルの集計に関する項目を参照してください。

オブジェクトのアクセスレベルについての詳細は、『SAP Business Objects Business Intelligence platform 管理者ガイド』を参照してください。

## 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)

- ・ 201 ページの[セキュリティプロファイルの集計オプションの変更](#)」
- ・ 189 ページの[ビジネスセキュリティプロファイルの表示データ設定](#)」

## 12.4.8 フィルタ設定の集計

ユニバースの 2 つ以上のビジネスセキュリティプロファイルが、同じユーザに割り当てられているか、同じユーザによって継承されている場合、ユーザがクエリやレポートを実行するとき、フィルタ設定の集計とクエリスクリプトに追加されるフィルタの決定には、次のルールが使用されます。

リレーショナルユニバースの場合、制限レベルに従ってフィルタが集計されます。最終的なフィルタには、クエリに適用される WHERE 句が追加されます。

制限レベル	集計ルール
非常に厳しい制限(デフォルト)	すべてのプロファイルのフィルタが、AND 演算子を使用して結合されます。
中程度の制限	継承されたフィルタが、AND 演算子を使用して集計されます。 結合されたフィルタが、OR 演算子を使用して集計されます。
緩い制限	すべてのプロファイルのフィルタが、OR 演算子を使用して結合されます。

OLAP ユニバースの場合、制限レベルに従って名前付きセットが集計されます。

制限レベル	集計ルール
非常に厳しい制限(デフォルト)	すべてのプロファイルで定義されたすべての名前付きセットに含まれるメンバーだけが表示されます。
中程度の制限	継承されたプロファイルで定義されたすべての名前付きセットに含まれ、結合されたプロファイルで定義された名前付きセットの少なくとも 1 つに含まれるメンバーが表示されます。
緩い制限	いずれかのプロファイルのいずれかの名前付きセットに含まれるメンバーが表示されます。

### 注

継承したプロファイルおよび結合されたプロファイルの定義については、セキュリティプロファイルの集計に関する項目を参照してください。

## 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)」
- ・ 201 ページの[セキュリティプロファイルの集計オプションの変更](#)」
- ・ 190 ページの[ビジネスセキュリティプロファイルのフィルタ設定](#)」

## 12.5 セキュリティエディタの使用

セキュリティエディタを使用してセキュリティプロファイルを作成および編集し、ユーザにプロファイルを割り当てます。




### 注

セキュリティは .unx universes のみに作成できます。

セキュリティエディタのタブにセッション名が表示されます。セッション名の先頭にアスタリスクが表示される場合は、リポジトリにまだ保存されていない、セキュリティプロファイルに対する変更や割り当てがセキュリティエディタで行われたことを意味します。

セキュリティエディタは、ユニバースごと、またはユーザ/グループの 2 つの方法で表示できます。セキュリティエディタの左側のタブを選択して、作業するビューを表示します。

- ・ [ユニバース/プロファイル] タブでは、まずリポジトリでユニバースを選択することによりタスクを実行できます。
- ・ [ユーザ/グループ] タブでは、最初にユーザまたはグループを選択して操作を実行できます。[ユーザ/グループ] パネルにある 3 つのアイコンを使用すると、次の 3 つの方法でユーザおよびグループを表示できます。

アイコン	説明
	ユーザのみを表示します。
	すべてのグループとそれに含まれるユーザを表示します。割り当てられたグループやユーザを持たないグループも表示されます。グループはフラットリストとして表示されます。  これは、デフォルト表示です。
	すべてのグループとそれに含まれるすべてのグループとユーザを表示します。グループは、そのさまざまな親グループと共に表示されます。


セントラル管理コンソールで付与されるアプリケーション権限によって、セキュリティエディタで実行できる操作内容が制御されます。詳細については、『SAP Business Objects Business Intelligence platform 管理者ガイド』の「アクセス権に関する付録」を参照してください。

セキュリティエディタを使用したセキュリティの作成の概要、または操作に関するヘルプについては、関連項目を参照してください。

#### 関連項目

- ・ 177 ページの[ユニバースセキュリティの概要](#)
- ・ 200 ページの[セキュリティプロファイルの挿入と編集](#)
- ・ 201 ページの[データセキュリティプロファイルの優先順位の変更](#)
- ・ 201 ページの[セキュリティプロファイルの集計オプションの変更](#)
- ・ 202 ページの[ユーザへのセキュリティプロファイルの割り当て](#)
- ・ 202 ページの[ユーザに割り当てられているプロファイルの表示とネットプロファイルのプレビュー](#)

## 12.5.1 セキュリティエディタを開く

- 1 インフォメーションデザインツールツールバーで、[セキュリティエディタ] アイコン  をクリックします。
- 2 [セッションを開く] ダイアログボックスで、開くセッションを選択します。
- 3 選択したセッションにまだログインしていない場合は、要求される情報を入力します。

セキュリティエディタが新しいタブで開きます。

#### 注

セキュリティエディタでは 1 度に複数のセッションを開くことができます。セッションは、異なるリポジトリにある必要があります。

#### 関連項目

- ・ 175 ページの[セッションを開く](#)
- ・ 199 ページの[セキュリティエディタの使用](#)

## 12.5.2 セキュリティプロファイルの挿入と編集

- 1 セキュリティエディタの [ユニバース/プロファイル] パネルで、ユニバースを選択します。
- 2 次のいずれかを実行します。
  - ・ プロファイルを挿入するには、ユニバース名を右クリックして、[データセキュリティプロファイルの挿入] または [ビジネスセキュリティプロファイルの挿入] を選択します。
  - ・ プロファイルを編集するには、プロファイル名をダブルクリックします。
- 3 編集するタブをクリックして、各タブのセキュリティ設定を定義します。

各タブに入力する情報のタイプの詳細については、ヘルプアイコンをクリックしてください。



**注**

[リセット] ボタンをクリックすると、すべてのタブの設定が、データファンデーションおよびビジネスレイヤで定義されているデフォルトの値に戻ります。

- 4 すべての設定を定義したら、[OK] をクリックします。
- 5 リポジトリにセキュリティ設定の変更を保存するには、メインツールバーの [保存] アイコンをクリックします。

**関連項目**

- ・ 200 ページの [セキュリティエディタを開く](#)

### 12.5.3 データセキュリティプロファイルの優先順位の変更

- 1 セキュリティエディタの [ユニバース/プロファイル] パネルで、ユニバースを選択します。
- 2 ユニバース名を右クリックして、[データセキュリティプロファイルの優先順位の変更] を選択します。

**注**

このコマンドは、ユニバースに複数のデータセキュリティプロファイルが定義されている場合にのみ使用できます。

- 3 データセキュリティプロファイルが一覧表示されるダイアログボックスで、矢印ボタンを使用して一覧の中でプロファイルを上下に移動します。一覧の最初にあるプロファイルが優先順位の最も高いプロファイルとなります。
- 4 優先順位の設定が完了したら、[OK] をクリックします。
- 5 リポジトリに変更を保存するには、メインツールバーの保存アイコンをクリックします。

**関連項目**

- ・ 191 ページの [セキュリティプロファイルの集計](#)
- ・ 200 ページの [セキュリティエディタを開く](#)

### 12.5.4 セキュリティプロファイルの集計オプションの変更

- 1 セキュリティエディタの [ユニバース/プロファイル] パネルで、ユニバースを選択します。  
ユニバースの現在の集計オプションは、エディタの右下に表示されます。
- 2 各セキュリティ設定について、ドロップダウンリストを使用して新しい集計オプションを選択します。  
このオプションは、現在選択されているユニバースにのみ適用されます。
- 3 リポジトリに変更を保存するには、メインツールバーの保存アイコンをクリックします。


## 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)
- ・ 200 ページの[セキュリティエディタを開く](#)

## 12.5.5 ユーザへのセキュリティプロファイルの割り当て

- 1 セキュリティエディタの [ユニバース/プロファイル] パネルで、ユニバースを選択します。  
現在割り当てられているすべてのユーザとグループが、[割り当てられたユーザ] 一覧に表示されます。
- 2 割り当てするには、エディタの右側のユーザー一覧でユーザまたはグループを選択し、[割り当てられたユーザ] 一覧を指す矢印をクリックします。
- 3 割り当てを解除するには、[割り当てられたユーザ] 一覧でユーザまたはグループを選択し、すべてのユーザの一覧を指す矢印をクリックします。

**警告**

二重矢印アイコン () は、ユーザおよびグループの選択の有無にかかわらず、すべてのユーザおよびグループの割り当てを解除します。

- 4 リポジトリに変更を保存するには、メインツールバーの保存アイコンをクリックします。

## 関連項目

- ・ 200 ページの[セキュリティエディタを開く](#)

## 12.5.6 ユーザに割り当てられているプロファイルの表示とネットプロファイルのプレビュー

- 1 セキュリティエディタで、エディタの左側の [ユーザ/グループ] パネルをクリックします。
- 2 [ユーザ/グループ] パネルで、ユーザまたはグループを選択します。
- 3 エディタの右上の [ユニバース/プロファイル] パネルで、ユニバースを選択します。

**ヒント**

[選択したユーザ/グループに割り当てられているユニバースのみ表示] チェックボックスを選択すると、選択したユーザに割り当てられているプロファイルを持つユニバースだけを一覧表示するように、表示を変更できます。

ユーザとユニバースを選択すると、エディタの右下のプロファイル一覧に、割り当てられているプロファイルが表示されます。

- 4 ネットデータセキュリティプロファイルやネットビジネスセキュリティプロファイルをプレビューするには、対応するプロファイル一覧の下に [ネットプロファイルのプレビュー] をクリックします。

データセキュリティプロファイルエディタかビジネスセキュリティプロファイルエディタが、読み取り専用モードで開きます。各タブに表示される設定では、そのユーザに割り当てられているすべてのプロファイルを集計した後使用されることになる設定が考慮されています。

#### 関連項目

- ・ 191 ページの[セキュリティプロファイルの集計](#)」
- ・ 200 ページの[セキュリティエディタを開く](#)」



## 関数のリファレンス

@ 関数は、オブジェクトにクエリスクリプトを指定するときにさらにフレキシブルなメソッドを提供する特別な関数です。SQL および MDX 式エディタ内の [関数] ボックスには、ユーザが定義する式で利用できる @関数が一覧表示されます。

以下は、インフォメーションデザインツール でサポートされている @関数の一覧です。@関数についての完全な説明は、『ユニバースデザインツールユーザガイド』を参照してください。

@ 関数	説明
@Aggregate_Aware	まず、@Aggregate_Aware 関数でパラメータとして指定されたすべての集計テーブルにクエリが実行されます。
@DerivedTable	派生テーブルの参照ネストされた派生テーブル（「派生テーブルの派生テーブル」とも呼ばれています）は、最低 1 つの既存の派生テーブルから派生したテーブルです。
@Prompt	<p>@Prompt 関数を使用するオブジェクトをクエリに含めるときに、ユーザに値を入力するよう指示するプロンプトが表示されます。</p> <p><b>注</b></p> <p>@Prompt 関数の既存の構文は、インフォメーションデザインツールでサポートされます。また、プロンプトの名前付きパラメータを定義し、@Prompt 関数を使用してクエリスクリプト内のパラメータを参照できます。たとえば以下のとおりです。</p> <p>@Prompt(&lt;パラメータ名&gt;)</p> <p>詳細については、パラメータに関する関連項目を参照してください。</p>
@Select	別のオブジェクトの SELECT 句を使用できるようになります。

@ 関数	説明																				
@Variable	システム変数に割り当てられた値を呼び出すために WHERE 句で使用されます。利用可能な変数は、次のとおりです。																				
	<table><tr><th>変数</th><th>説明</th></tr><tr><td>BOUSER</td><td>ユーザのログイン名</td></tr><tr><td>DBUSER</td><td>データソースに接続するときの認証に使用される名前。</td></tr><tr><td>UNVID</td><td>ユニバースの ID</td></tr><tr><td>UNVNAME</td><td>ユニバースの名前</td></tr><tr><td>PREFERRED_VIEWING_LOCALE</td><td>アプリケーションでレポートとクエリオブジェクトを表示するためのユーザの優先ロケール。</td></tr><tr><td>DOMINANT_PREFERRED_VIEWING_LOCALE</td><td>フォールバックリソースがリソースに定義されていない場合に使用される事前定義済みのフォールバックロケール。</td></tr><tr><td>DPTYPE</td><td>データプロバイダの種類</td></tr><tr><td>DPNAME</td><td>データプロバイダ名</td></tr><tr><td>DOCNAME</td><td>ドキュメント名</td></tr></table>	変数	説明	BOUSER	ユーザのログイン名	DBUSER	データソースに接続するときの認証に使用される名前。	UNVID	ユニバースの ID	UNVNAME	ユニバースの名前	PREFERRED_VIEWING_LOCALE	アプリケーションでレポートとクエリオブジェクトを表示するためのユーザの優先ロケール。	DOMINANT_PREFERRED_VIEWING_LOCALE	フォールバックリソースがリソースに定義されていない場合に使用される事前定義済みのフォールバックロケール。	DPTYPE	データプロバイダの種類	DPNAME	データプロバイダ名	DOCNAME	ドキュメント名
	変数	説明																			
	BOUSER	ユーザのログイン名																			
	DBUSER	データソースに接続するときの認証に使用される名前。																			
	UNVID	ユニバースの ID																			
	UNVNAME	ユニバースの名前																			
	PREFERRED_VIEWING_LOCALE	アプリケーションでレポートとクエリオブジェクトを表示するためのユーザの優先ロケール。																			
	DOMINANT_PREFERRED_VIEWING_LOCALE	フォールバックリソースがリソースに定義されていない場合に使用される事前定義済みのフォールバックロケール。																			
	DPTYPE	データプロバイダの種類																			
DPNAME	データプロバイダ名																				
DOCNAME	ドキュメント名																				
@Where	別のオブジェクトの WHERE 句を使用できるようになります。																				

以下の表では、異なるクエリ式で使える @関数を示します。

@ 関数	結合	計算列	派生テーブル	ビジネスオブジェクト
@Aggregate_Aware	許可されていない	許可されていない	許可されていない	許可されている (SQL のみ)

@ 関数	結合	計算列	派生テーブル	ビジネスオブジェクト
@DerivedTable	許可されていない	許可されていない	許可されている <b>注</b> データベース固有の SQL (複数のソースが有効化されたデータファンデーション) では、すべての引数は同じ接続からのテーブルまたは列を参照する必要があります。	許可されていない
@Prompt	許可されている	許可されている <b>注</b> 複数のソースが有効化されたデータファンデーション内のデータベース固有の SQL では使用できません。	許可されている	許可されている
@Select	許可されていない	許可されていない	許可されていない	許可されている
@Variable	許可されている	許可されている <b>注</b> データベース固有の SQL (複数のソースが有効化されたデータファンデーション) では、すべての引数は同じ接続からのテーブルまたは列を参照する必要があります。	許可されている	許可されている
@Where	許可されていない	許可されていない	許可されていない	許可されている (SQL のみ)

## 関連項目

- ・ 124 ページの [パラメータについて](#)」





## SQL 生成パラメータのリファレンス

### 14.1 SQL 生成パラメータについて

SQL 生成パラメータは、クエリスクリプトの生成に影響を与えます。すべてのパラメータにはデフォルトの値があります。デフォルト値はデータファンデーションのプロパティで上書きできます。ビジネスレイヤのプロパティで上書きできるパラメータ (値の一覧に関する) もあります。クエリ実行時には、クエリサーバは、以下の順番で特定された値を使用します。

- 1 設定されている場合は、ビジネスレイヤの値。
- 2 設定されている場合は、データファンデーションの値。
- 3 デフォルト値

#### 関連項目

- ・ 209 ページの[SQL 生成パラメータのリファレンス](#)」
- ・ 93 ページの[データファンデーションのプロパティについて](#)」
- ・ 104 ページの[ビジネスレイヤのプロパティについて](#)」

### 14.2 SQL 生成パラメータのリファレンス

以下のリファレンスは、データファンデーションプロパティおよびビジネスレイヤプロパティで上書きできる SQL 生成パラメータについて説明しています。

#### 14.2.1 ANSI92

ANSI92 = Yes|No

値	Yes/No
---	--------

デフォルト	No
説明	生成された SQL が ANSI92 標準に準拠するかどうかを指定します。 Yes: ANSI92 標準に準拠した SQL 生成を有効にします。 No: PRM パラメータ OUTER_JOIN_GENERATION に従って SQL を生成します。

### 14.2.2 AUTO\_UPDATE\_QUERY

AUTO\_UPDATE\_QUERY = Yes|No

値	Yes/No
デフォルト	No
説明	クエリのオブジェクトがユーザプロファイルで利用できない場合の動作を指定します。 Yes: クエリが更新され、オブジェクトがクエリから削除されます。 No: オブジェクトがクエリに残されます。

### 14.2.3 BEGIN\_SQL

BEGIN\_SQL = <文字列>

値	文字列
デフォルト	空の文字列

説明	<p>会計、優先順位付け、負荷管理に使用する SQL 文のプレフィックスとして使用します。このパラメータは、ドキュメント生成、LOV クエリを含むすべての SQL 生成に適用されます。</p> <p>このパラメータは、Web Intelligence、LiveOffice、および QaaWS でサポートされています。しかし、Desktop Intelligence および Crystal Reports では無視されます。</p> <p>Teradata の例</p> <pre>BEGIN_SQL=SET QUERY_BAND='string' for transaction;</pre> <p>このパラメータには、1 つ以上の名前と値のペアが含まれる文字列が必要です。これらのペアは、セミコロンで区切られ、一重引用符で囲まれています。すべての SQL 文は、BEGIN_SQL に続くこのパラメータがプレフィックスとして付いています。このパラメータに入力される名前と値のペアは、GetQueryBandPairs システムテーブルに入力されます。</p> <p>3 つの名前と値のペアの例:</p> <pre>BEGIN_SQL=SET QUERY_BAND='UserID=Jones;JobID=980;AppID=TRM' for transaction;</pre> <p>@Variable 関数を名前と値のペアの値として使用することもできます。戻り値は、一重引用符で囲まれます。BEGIN_SQL=SET QUERY_BAND='USER=@Variable('BOUSER');Document=@Variable('DPNAME');' for transaction;</p>
----	--

## 14.2.4 BLOB\_COMPARISON

BLOB\_COMPARISON = Yes|No

値	Yes/No
デフォルト	No
編集の可/不可	不可

説明	<p>BLOB ファイルが DISTINCT 句で使用されている場合、クエリを SELECT 句とともに生成するかどうかを指定します。クエリプロパティの No Duplicate Row の設定に関連しています。</p> <p>Yes: DISTINCT 句をクエリ内で使用できます。</p> <p>No: クエリ設定の No Duplicate Row がオンになっていても、クエリ内で DISTINCT 文を使用できません。</p>
----	--

## 14.2.5 BOUNDARY\_WEIGHT\_TABLE

BOUNDARY\_WEIGHT\_TABLE = Integer 32bits [0-9]

値	Integer 32bits [0 ~ 9、または負の整数]
デフォルト	-1
説明	<p>テーブルに多くの行がある場合、FROM 句の最適化を可能にします。</p> <p>テーブルのサイズ(行数)が入力値より大きい場合、テーブルはサブクエリとして宣言されます。</p> <p>FROM (SELECT col1, col2,....., coln, ....., FROM Table_Name WHERE simple condition)</p> <p>簡単な条件は、サブクエリをもたないものとして定義されます。</p> <p>-1、0、または任意の負の数は、この最適化が使用されていないことを意味します。</p>
制限	<p>最適化は以下の場合実装できません。</p> <ul style="list-style-type: none"><li>・ OR 演算子がクエリ条件にある。</li><li>・ 1 つのテーブルだけが SQL に関係している。</li><li>・ クエリに外部結合が入っている。</li><li>・ 最適化するテーブルで条件が指定されていない。</li><li>・ 最適化するテーブルが派生テーブルである。</li></ul>

## 14.2.6 COLUMNS\_SORT

COLUMNS\_SORT = Yes|No

値	Yes/No
デフォルト	No
説明	列を[構造]枠のテーブルに表示する順序を指定します。 Yes: 列は文字順に表示されます。 No: 列はデータベースから取得された順序で表示されます。

## 14.2.7 CUMULATIVE\_OBJECT\_WHERE

CUMULATIVE\_OBJECT\_WHERE = Yes|No

値	Yes No
デフォルト	No

説明	<p>このパラメータは、フィルタされたオブジェクトのみに適用されます。WHERE 句とこれらのオブジェクトのクエリ条件を組み合わせる方法を指定します。</p> <p>Yes: WHERE 句が、メインクエリの条件と AND 演算子で組み合わせられるように指定します。</p> <p>No: オブジェクトの WHERE 句が、このオブジェクトの条件と組み合わせられるように指定します。</p> <p>例</p> <p>John ではないフランス人のクライアントすべて、またはニューヨークではないアメリカの都市を探す場合、SQL は次のようになります。</p> <p>Yes:</p> <pre>(customer.first_name &lt;&gt; 'John') OR (city.city &lt;&gt; 'New York' AND customer_country.country = 'France' AND city_country.country = 'USA')</pre> <p>No:</p> <pre>(customer.first_name &lt;&gt; 'John' AND customer_country.country = 'France' ) OR (city.city &lt;&gt; 'New York' AND city_country.country = 'USA' )</pre>
----	---

## 14.2.8 DISABLE\_ARRAY\_FETCH\_SIZE\_OPTIMIZATION

DISABLE\_ARRAY\_FETCH\_SIZE\_OPTIMIZATION = Yes|No

値	Yes/No
デフォルト	No

説明	<p>最適化アルゴリズムを使用して、デフォルト設定を使用せずに、返される配列のサイズを最適化することができます。</p> <p>No: ユニバースで実行されるすべてのクエリが、最適化によるメリットを享受します。</p> <p>Yes: クエリは、デフォルト値のセットを使用します。</p>
----	--

## 14.2.9 DISTINCT\_VALUES

DISTINCT\_VALUES = GROUPBY|DISTINCT

値	GROUPBY DISTINCT
デフォルト	DISTINCT
説明	<p>[重複行は無視する]オプションが有効な場合、SQL を DISTINCT または GROUPBY 句と共に値の一覧やクエリパネルに生成するかどうかを指定します。</p> <p>DISTINCT: SQL が DISTINCT 句とともに生成されます。次はその例です。</p> <p>SELECT DISTINCT cust_name FROM Customers</p> <p>GROUPBY: SQL が GROUP BY 句とともに生成されます。次はその例です。</p> <p>SELECT cust_name FROM Customers GROUP BY cust_name</p>

## 14.2.10 END\_SQL

END\_SQL = 文字列

値	文字列
---	-----

デフォルト	<空の文字列>
説明	このパラメータで指定した文が、各 SQL 文の末尾に追加されます。
例	<p>IMB DB2 データベースで、次のように使用できます。</p> <p>END_SQL=FOR SELECT ONLY</p> <p>サーバで、データブロックの読み取りが高速で行われます。</p> <p>次の例は別の宣言方法です。</p> <p>END_SQL='write ' UNVID To Usage_Audit.Querieded_universe</p> <p>監査テーブルにユニバース ID が書き込まれます。これは、ユーザ、クエリされたテーブルなど他のデータを記録するために使用されます。</p>

#### 14.2.11 EVAL\_WITHOUT\_PARENTHESES

EVAL\_WITHOUT\_PARENTHESES = Yes|No

値	Yes No
デフォルト	No



説明	<p>デフォルトで、@Select(クラス¥オブジェクト) 関数が、かっこでくられた〈クラス¥オブジェクト〉オブジェクトの SELECT 句で置き換えられます。</p> <p>たとえば、2 つの @Select 句を組み合わせた @Select(object1)*@Select(object2) は、</p> <p>SQL(object1) = A-B で SQL(object2) =C とすると、</p> <p>( A-B ) * ( C ) となります。</p> <p>デフォルトではかっこが追加されますが、EVAL_WITHOUT_PARENTHESIS = Yes に設定することによってこれを回避できます。この場合、A-B * C となります。</p> <p>Yes: かっこが SELECT(クラス¥オブジェクト) 関数の @Select 句から削除されます。</p> <p>No: かっこが @Select(クラス¥オブジェクト) 関数の SELECT 句を囲むように追加されます。</p>
----	--

#### 14.2.12 FORCE\_SORTED\_LOV

FORCE\_SORTED\_LOV = Yes|No

値	Yes No
デフォルト	No
説明	<p>並べ替えられた値の一覧を取得します。</p> <p>Yes: 値の一覧を並べ替えられるよう指定します。</p> <p>No: 値の一覧を並べ替えないよう指定します。</p>

#### 14.2.13 INNERJOIN\_IN\_WHERE

INNERJOIN\_IN\_WHERE = Yes|No

値	Yes No
デフォルト	No。パラメータを有効にするには、そのパラメータを手動で追加する必要があります。
説明	<p>ANSI92 が Yes に設定されている場合、SQL 構文がすべての内部結合と共に WHERE 句に強制的に生成されるよう指定できます。この設定は、クエリに FULL OUTER、RIGHT OUTER、または LEFT OUTER 結合がなく、内部結合だけが含まれる場合に限り有効です。</p> <p>Yes: クエリに内部結合だけが含まれる場合を除き、ANSI92 が Yes に設定されると、ANSI92 結合構文が FROM 句に生成されます。この場合、内部結合は WHERE 句に含まれます。</p> <p>No: ANSI92 が Yes に設定されている場合、ANSI 句に FROM 92 結合構文が生成されます。</p>

#### 14.2.14 JOIN\_BY\_SQL

JOIN\_BY\_SQL = Yes|No

値	Yes No
デフォルト	No
説明	<p>複数の SQL 文をどのように処理するかを指定します。データベースで許可されている場合に限り、複数の SQL 文を組み合わせることができます。</p> <p>Yes: 複数の SQL 文が組み合わせられることを指定します。</p> <p>No: 複数の SQL 文が組み合わせられないことを指定します。これはデフォルト値です。</p>

#### 14.2.15 MAX\_INLIST\_VALUES

MAX\_INLIST\_VALUES = [0-99]

値	整数: 最小 -1、最大は DB に依存
デフォルト	-1
説明	<p>IN LIST 演算子を使用する場合、条件に入力できる値の最大数を設定することができます。</p> <p>99 : IN LIST 演算子を使用して条件を作成するとき、値を 99 個まで入力できるよう指定します。</p> <p>入力できる最大値はデータベースに依存します。</p> <p>値が -1 の場合は、データベースによる制限を除いて、返される値の数に制限はありません。</p>

#### 14.2.16 REPLACE\_COMMA\_BY\_CONCAT

REPLACE\_COMMA\_BY\_CONCAT= Yes|No

値	Yes No
デフォルト	No
説明	<p>旧バージョンのユニバースデザインツールでは、オブジェクトの SELECT 句の複数フィールドを区切るのにカンマを使用することができました。カンマは連結演算子として処理されました。すでにカンマがこのように使われているユニバースの場合、REPLACE_COMMA_BY_CONCAT を No に設定してこの動作を維持できます。現バージョンのユニバースデザインツールでは、デフォルトでこのパラメータが Yes に設定されます。その結果、カンマを複数フィールドの区切りとして使用する式は、自動的に連結構文を使用するよう設定されます。</p> <p>Yes: 複数フィールドオブジェクトがある場合、カンマが連結式に置き換えられます。</p> <p>No: カンマをそのままにしておきます。</p>

## 14.2.17 SELFJOINS\_IN\_WHERE

SELFJOINS\_IN\_WHERE = Yes|No

値	Yes No
デフォルト	No
説明	<p>自己結合は通常、FROM 句に含まれます。このパラメータにより、自己結合のすべての条件を WHERE 句に指定した SQL 構文を生成することができます。このパラメータが考慮されるようにするには、ANSI92 パラメータを Yes に設定する必要があります。</p> <p>パラメータを有効にするには、そのパラメータを手動で一覧に追加する必要があります。</p> <p>Yes: 自己結合の条件は、SQL クエリの WHERE 句に含まれます。</p> <p>No: 自己結合の構文は ANSI 規格に基づいて生成され、自己結合の条件は SQL クエリの ON 句のテーブル結合定義の FROM 句に含まれます。</p>

## 14.2.18 SHORTCUT\_BEHAVIOR

SHORTCUT\_BEHAVIOR = ShortestPath|Global|Successive

値	ShortestPath Global Successive
デフォルト	ShortestPath

説明	<p>ショートカット結合の適用方法を指定します。</p> <p>ShortestPath: クエリで取得するテーブル数が最小になるようにショートカットを適用します。</p> <p>Successive: ショートカットを連続で適用します。ショートカットによりその後の連続ショートカットに含まれるテーブルが削除されると、連続ショートカットは適用されません。</p> <p>Global: すべてのショートカットを適用します。クエリによりデカルト積が生成される場合、ショートカット結合は適用されません。</p> <p><b>注</b> このパラメータは、以前 GLOBAL_SHORTCUTS PRM ファイルで としてリストされていました。値は、Global が Yes に相当し、Successive が No. に相当します。</p>
----	--

#### 14.2.19 SMART\_AGGREGATE

SMART\_AGGREGATE = Yes|No

値	Yes No
デフォルト	No
説明	<p>集計テーブルに基づくスマートメジャーに対する集計テーブルの使用方法を決定します。これにより、比率に基づくユニバースオブジェクトが正しく集計されるようになります。デフォルトでは、集計テーブルの計算済みの値を利用するため、これらのテーブルがすべての期間(異なる期間)を通じて整合性がない場合は、このパラメータを使用して最も詳細な集計テーブルが使用されるようにしてください。</p> <p>このパラメータは、ユニバースパラメータリストには表示されず、デフォルトでは無効になります。ユニバース設計者は、このパラメータを有効(値 Yes )にする前にパラメータリストに手動で挿入する必要があります。</p> <p>Yes: 追加のグループ設定クエリは、集計テーブルに基づくスマートメジャーに対する最初のクエリの集計テーブルを基盤とします。</p> <p>No: システムは、最も適切な集計テーブルを選択します。</p>

## 14.2.20 THOROUGH\_PARSE

THOROUGH\_PARSE = Yes|No

値	Yes No
デフォルト	No
説明	<p>クエリパネルのデフォルトの分析および個々のオブジェクトの分析に使用する手法を指定します。</p> <p>Yes: PREPARE, DESCRIBE および EXECUTE 文を使用して、オブジェクトの SQL を分析します。</p> <p>Prepare+DescribeCol+Execute</p> <p>No: PREPARE および DESCRIBE 文を使用して、オブジェクトの SQL を分析します。</p>

## 14.2.21 TRUST\_CARDINALITIES

TRUST\_CARDINALITIES = Yes|No

値	Yes No
デフォルト	No
説明	<p>結果が増大した場合に SQL を最適化できます。</p> <p>Yes: メジャーを含むクエリの場合は、メジャーを増大させ、また[結果オブジェクト]に表示されないすべての条件が、サブクエリに変換されます。これによって、メジャーに対して False の結果を返す可能性があるテーブルはクエリに含まれなくなります。</p> <p>No: 最適化は実装されません。</p>

## 14.2.22 UNICODE\_STRINGS

UNICODE\_STRINGS = Yes|No

値	Yes No
デフォルト	No
説明	<p>現在のユニバースで Unicode 文字列を操作できるかどうかを指定します。Microsoft SQL Server および Oracle 9 にのみ適用されます。SBO ファイルでデータベースの文字セットが Unicode に設定されていれば、SQL の生成を修正し、NCHAR と NVARCHAR のような特定の Unicode 列の型を処理できるようにする必要があります。</p> <p>Yes: 文字列に基づく条件が、UNICODE_PATTERN ファイルのパラメータ PRM の値に従って SQL で書式設定されます。たとえば、MS SQL Server(sql srv.prm) の場合、UNICODE_PATTERN=N\$ となります。</p> <p>条件 Customer_name='Arai' は</p> <p>Customer_name=N'Arai' となります。</p> <p>注: Unicode 値を基に @Prompt 構文を使って新しいプロンプトを作成すると、データ型は 'C' ではなく 'U' になります。</p> <p>No: 文字列に基づくすべての条件は、標準の SQL の書式で表現されます。たとえば、Customer_name='Arai' はそのまま Customer_name='Arai' となります。</p>





## SQL 関数リファレンス

以下のリファレンスは、複数のソースが有効化されたデータファンデーションとそれに依存するビジネスレイヤで SQL 式を定義するときに使用できる SQL 関数について説明しています。

### 15.1 集計関数

集計関数は、データセットに対して演算を行います。

集計関数では、列名の前で SQL キーワード `distinct` を使用できます。

#### 15.1.1 AVG

値のセットの平均値を返します。

構文	DECIMAL AVG(INTEGER n)  DECIMAL AVG(DECIMAL d)
例	<ul style="list-style-type: none"><li>INTEGERS または DECIMALS の型の値を含む 2 つの列の合計を平均する  = AVG( S1.A1 + S1.A2 )</li><li>STRINGS の型で書き込まれた数値を含む列内の、複数の値を平均する  = AVG( toInteger( S1.A1 ) )</li></ul>

15.1.2 COUNT

セット内の値の個数をカウントします。

構文	INTEGER COUNT(INTEGER n)
	INTEGER COUNT(DECIMAL c)
	INTEGER COUNT(DOUBLE d)
	INTEGER COUNT(String s)
	INTEGER COUNT(TIMESTAMP m)
	INTEGER COUNT(TIME t)
	INTEGER COUNT(DATE a)
	INTEGER COUNT(BOOLEAN b)
例	<div>・ 単一系列内の値の個数をカウントする</div> <div>= COUNT( S1.A1 )</div>

15.1.3 MAX

セット内の最大値を返します。

構文	<div>INTEGER MAX(INTEGER n)</div> <div>DECIMAL MAX(DECIMAL c)</div> <div>DOUBLE MAX(DOUBLE d)</div> <div>STRING MAX(STRING s)</div> <div>TIMESTAMP MAX(TIMESTAMP m)</div> <div>TIME MAX(TIME t)</div> <div>DATE MAX(DATE d)</div>
例	<div>・ 単一系列内の最大値を返す</div> <div>= MAX( S1.A1 )</div>

#### 15.1.4 MIN

セット内の最小値を返します。

構文	<div>INTEGER MIN(INTEGER n)</div> <div>DECIMAL MIN(DECIMAL c)</div> <div>DOUBLE MIN(DOUBLE d)</div> <div>STRING MIN(STRING s)</div> <div>TIMESTAMP MIN(TIMESTAMP m)</div> <div>TIME MIN(TIME t)</div> <div>DATE MIN(DATE d)</div>
例	<div>・ 単一系列内の最小値を返す</div> <div>= MIN( S1.A1 )</div>

15.1.5 SUM

値のセットの合計値を返します。

構文	<div>DECIMAL SUM(INTEGER n)</div> <div>DECIMAL SUM(DECIMAL c)</div> <div>DECIMAL SUM(DOUBLE d)</div>
例	<div>・ 単一系列内の値の合計値を計算する</div> <div>= SUM( S1.A1 )</div>

15.2 数値関数

15.2.1 abs

数値引数の正の絶対値を返します。

構文	<div>decimal abs(DECIMAL n)</div> <div>double abs(DOUBLE n)</div> <div>integer abs(INTEGER n)</div>
制約事項	<div><math>\text{abs}(-2^{31}) = -2^{31}</math></div> <div>引数が NULL の場合、NULL を返します。</div>

### 15.2.2 acos

0 から  $\pi$  の範囲で指定した角度の逆余弦値を返します。

構文	<code>double acos(DOUBLE d)</code>
制約事項	<code>abs(d) &gt; 1</code> の場合、NULL を返します。

### 15.2.3 asin

$-\pi/2$  から  $\pi/2$  の範囲で指定した角度の逆正弦値を返します。

構文	<code>double asin(DOUBLE d)</code>
制約事項	<code>abs(d) &gt; 1</code> の場合、NULL を返します。

### 15.2.4 atan

$-\pi/2$  から  $\pi/2$  の範囲で指定した角度の逆正接値を返します。

構文	<code>double atan(DOUBLE d)</code>
----	------------------------------------

### 15.2.5 atan2

`atan2(x,y)` は、直交座標  $(x,y)$  を極座標  $(r, \theta)$  に変換します。この方法は、 $-\pi$  から  $\pi$  となる  $y/x$  の逆正接値を計算することによって、位相  $\theta$  を求めます。

構文	<code>double atan2(DOUBLE x, DOUBLE y)</code>
制約事項	$x = 0$ かつ $y = 0$ の場合、NULL を返します。

### 15.2.6 ceiling

引数以上で数学的整数となる最小の値を返します。

構文	<code>integer ceiling(INTEGER n)</code> <code>double ceiling(DOUBLE n)</code> <code>decimal ceiling(DECIMAL n)</code>
----	---

### 15.2.7 cos

指定した角度の余弦値を返します。

構文	<code>double cos(DOUBLE d)</code>
----	-----------------------------------

### 15.2.8 cot

指定した角度の余接値を返します。正弦値が 0 である場合は NULL を返します。

構文	<code>double cot(DOUBLE d)</code>
----	-----------------------------------

制約事項	$\sin(d) = 0$ の場合は NULL を返します。
------	--------------------------------

### 15.2.9 degrees

弧度法で表された角度を度数法で表された近似値に変換します。

構文	<code>double degrees(INTEGER n)</code> <code>double degrees(DOUBLE d)</code> <code>double degrees(DECIMAL c)</code>
----	---

### 15.2.10 exp

DOUBLE 値 d の指数関数値を返します。e を d 乗した数値となります。

構文	<code>double exp(DOUBLE d)</code>
例	<code>exp(10) = e<sup>10</sup> = 22 026.4658</code>
制約事項	オーバーフローの場合は例外処理を行います。

### 15.2.11 floor

引数以下で数学的整数となる最大の値を返します。

#### 注

返された値の型は変換されません。したがって、`floor(1.9) = 1.0` となります。値を INTEGER に変換するには、`toInteger()` などの変換関数を使用してください。

構文	<code>integer floor(INTEGER n)</code> <code>double floor(DOUBLE n)</code> <code>decimal floor(DECIMAL n)</code>
----	---

### 15.2.12 log

DOUBLE 値 d の e を底とする対数を返します。引数 d は 0 より大きい必要があります。引数が 0 または負の値である場合は NULL を返します。

構文	<code>double log(DOUBLE d)</code>
制約事項	d が 0 以下の場合、NULL を返します。

### 15.2.13 log10

DOUBLE 値 d の 10 を底とする対数を返します。引数 d は 0 より大きい必要があります。引数が 0 または負の値である場合は NULL を返します。

構文	<code>double log10(DOUBLE d)</code>
----	-------------------------------------

### 15.2.14 mod

INTEGER n1 を INTEGER n2 で割ったときの余りを返します。

構文	<code>integer mod(INTEGER n1, INTEGER n2)</code>
----	--



制約事項	n2 = 0 の場合は NULL を返します。
------	-------------------------

### 15.2.15 pi

定数  $\pi$  を返します。

構文	double pi()
----	-------------

### 15.2.16 power

指数乗した値を返します。指数は INTEGER である必要があります。

構文	double power(INTEGER n1, INTEGER n2) double power(DOUBLE n1, INTEGER n2) decimal power(DECIMAL n1, INTEGER n2)
制約事項	n1 = 0 かつ n2 < 0 の場合は NULL を返します。 オーバーフローの場合は例外処理を行います。

### 15.2.17 radians

度数法で表された角度を弧度法で表された近似値に変換します。

構文	<code>double radians(INTEGER n)</code> <code>double radians(DOUBLE d)</code> <code>double radians(DECIMAL c)</code>
----	---

### 15.2.18 rand

0 ≤ d < 1 となる DOUBLE 値 d を返します。種 (整数) を与えて乱数ジェネレータを初期化することができます。

構文	<code>double rand(INTEGER n)</code> <code>double rand()</code>
----	---

### 15.2.19 round

指定した小数点以下第 p 位までに丸めた値を返します。この関数は、切り上げた値との差と切り捨てた値との差が等しい場合を除いて、差が小さい値に丸めます。差が等しい場合は、切り上げを行います (0 から遠い方へ丸めます)。

p を指定しない場合、この関数は小数点以下なしの値への丸めを行います。

#### 注

返された値の型は変換されません。したがって、`round(1.9) = 2.0` となります。値を INTEGER に変換するには、`toInteger()` などの変換関数を使用してください。

構文	<code>integer round(INTEGER n, INTEGER p)</code> <code>double round(DOUBLE n, INTEGER p)</code> <code>decimal round(DECIMAL n, INTEGER p)</code> <code>integer round(INTEGER n)</code> <code>double round(DOUBLE n)</code> <code>decimal round(DECIMAL n)</code>
----	---

制約事項	この関数は、切り上げた値との差と切り捨てた値との差が等しい場合を除いて、差が小さい値に丸めます。差が等しい場合は 0 から遠い方へ丸めます。
------	--

### 15.2.20 sign

引数の符号である正 (1)、ゼロ (0)、負 (-1) を返します。

構文	<code>integer sign(INTEGER n)</code> <code>decimal sign(DECIMAL c)</code> <code>double sign(DOUBLE d)</code>
----	--

### 15.2.21 sin

指定した角度の正弦値を返します。

構文	<code>double sin(DOUBLE d)</code>
----	-----------------------------------

### 15.2.22 sqrt

数値の平方根を返します。引数は正である必要があります。引数が負の場合は NULL を返します。

構文	<code>double sqrt(DOUBLE d)</code>
制約事項	d が 0 より小さい場合は NULL を返します。

### 15.2.23 tan

指定した角度の正接値を返します。

構文	<code>double tan(DOUBLE d)</code>
制約事項	<code>cos(d) = 0</code> の場合は <code>NULL</code> を返します。

### 15.2.24 trunc

指定した `n` を小数点以下第 `m` 位までに切り捨てた値を返します。`m` を省略した場合 `n` は小数点以下なしの値へ切り捨てられます。

`m` が負の場合、この関数は小数点以上第 `m` 位での切り捨てを行い、その位置およびその右側にあるすべての数は `0` にセットされます。

構文	<code>integer trunc(INTEGER n, INTEGER m)</code> <code>double trunc(DOUBLE n, INTEGER m)</code> <code>decimal trunc(DECIMAL n, INTEGER m)</code> <code>integer trunc(INTEGER n)</code> <code>double trunc(DOUBLE n)</code> <code>decimal trunc(DECIMAL n)</code>
エイリアス	<code>truncate()</code>
例	<code>trunc( 10.1234, 1) = 10.1</code> <code>trunc( 10.1234, 2 ) = 10.12</code> <code>trunc( 1862.1234, -1 ) = 1860</code> <code>trunc( 1862.1234, -2 ) = 1800</code>

## 15.3 日付/時刻関数

### 15.3.1 curdate

戻り値: 現在の日付

date

現在の日付を日付値として返します。この関数はシステム関数で、以下の特徴があります。

- ・ 非決定的です。
- ・ データソースからではなく、データフェデレーションサービスからの値を返します。

### 15.3.2 curtime

戻り値: 現在の時間

time

ローカル時間を時間値として返します。この関数はシステム関数で、以下の特徴があります。

- ・ 非決定的です。
- ・ データソースからではなく、データフェデレーションサービスからの値を返します。

### 15.3.3 dayName

日付 *a* またはタイムスタンプ *m* の曜日を表す文字を返します。

構文	<pre>string dayName(date a) string dayName(timestamp m)</pre>
----	---

制約事項	<p>曜日名は大文字の英語表記で返されます。次のいずれかの値が返されます。</p> <ul style="list-style-type: none"><li>・ SUNDAY</li><li>・ MONDAY</li><li>・ TUESDAY</li><li>・ WEDNESDAY</li><li>・ THURSDAY</li><li>・ FRIDAY</li><li>・ SATURDAY</li></ul>
------	--

#### 15.3.4 dayNameL

ロケール l を使用して、日付 a または タイムスタンプ m の曜日要素を表す文字列を返します。

ロケール l には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。

構文	<pre>string dayNameL(date a, string l) string dayNameL(timestamp m, string l)</pre>
例	<pre>dayNameL(toTimestamp('2001-12-30 10:12:32.222'), 'en_US') = SUNDAY dayNameL(toTimestamp('2001-12-30 10:12:32.222'), 'fr_FR') = DIMANCHE dayNameL(toTimestamp('2001-12-30 10:12:32.222'), 'de_DE') = SONNTAG</pre>
制限	ラテンベースのスクリプトでは、名前が大文字で返されます。

#### 関連項目

- ・ 237 ページの [dayName](#)
- ・ 282 ページの [複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)

### 15.3.5 dayOfMonth

日付 *a* またはタイムスタンプ *m* の、日の部分を表す整数 1 ～ 31 を返します。

構文	<code>integer dayOfMonth(date a)</code> <code>integer dayOfMonth(timestamp m)</code>
----	---

### 15.3.6 dayOfWeek

日付 *a* またはタイムスタンプ *m* の、曜日を表す整数 1 ～ 7 を返します。週の最初の日は日曜です。

構文	<code>integer dayOfWeek(date a)</code> <code>integer dayOfWeek(timestamp m)</code>
制約事項	週の最初の日は日曜です。

### 15.3.7 dayOfWeekL

日付 *a* またはタイムスタンプ *m* の、曜日を表す整数 1 ～ 7 を返します。週の初めの曜日はロケール *l* に依存します。

ロケール *l* には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。

構文	<code>integer dayOfWeekL(date a, string l)</code> <code>integer dayOfWeekL(timestamp m, string l)</code>
----	---

例	<pre>dayOfWeekL(toTimestamp('2001-12-30 10:12:32.222'), 'en_US') = 1  dayOfWeekL(toTimestamp('2001-12-30 10:12:32.222'), 'fr_FR') = 7  dayOfWeekL(toTimestamp('2001-12-30 10:12:32.222'), 'de_DE') = 7  dayOfWeekL(toTimestamp('2001-12-30 10:12:32.222'), 'ja_JP') = 1</pre>
制限	<p>週の初めの曜日はロケール 1 に依存します。</p> <p>詳細については、Java API の Calendar クラス定義を参照してください。</p>

## 関連項目

- ・ 239 ページの [dayOfWeek](#)」
- ・ 282 ページの [複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)」
- ・ <http://java.sun.com/j2se/1.5.0/docs/api/java/util/Calendar.html>

### 15.3.8 dayOfYear

日付 a またはタイムスタンプ m の、通年日を表す整数 1 ～ 366 を返します。

構文	<pre>integer dayOfYear(date a)  integer dayOfYear(timestamp m)</pre>
----	--

### 15.3.9 decrementDays

日付 a またはタイムスタンプ m から、指定日数 n を引きます。



構文	<code>date decrementDays(date a, integer n)</code> <code>timestamp decrementDays(timestamp m, integer n)</code>
----	--

### 15.3.10 hour

時刻 `t` またはタイムスタンプ `m` の時のコンポーネントを示す整数 0 ～ 23 を返します。

構文	<code>integer hour(time t)</code> <code>integer hour(timestamp m)</code>
----	---

### 15.3.11 incrementDays

日付 `a` またはタイムスタンプ `m` に、指定日数 `n` をとる引数を加えます。

構文	<code>date incrementDays(date a, integer n)</code> <code>timestamp incrementDays(timestamp t, integer n)</code>
----	--

### 15.3.12 minute

時刻 `t` またはタイムスタンプ `m` の分のコンポーネントを示す整数 0 ～ 59 を返します。

構文	<code>integer minute(time t)</code> <code>integer minute(timestamp t)</code>
----	---

### 15.3.13 month

日付 a またはタイムスタンプ m の月のコンポーネントを示す整数 1 ～ 12 を返します。

構文	<code>integer month(date a)</code> <code>integer month(timestamp m)</code>
----	---

### 15.3.14 monthName

日付 a またはタイムスタンプ m の、月の部分を表す文字列を返します。

構文	<code>string monthName(date a)</code> <code>string monthName(timestamp m)</code>
制約事項	<p>曜日名は大文字の英語表記で返されます。次のいずれかの値が返されます。</p> <ul style="list-style-type: none"><li>・ JANUARY</li><li>・ FEBRUARY</li><li>・ MARCH</li><li>・ APRIL</li><li>・ MAY</li><li>・ JUNE</li><li>・ JULY</li><li>・ AUGUST</li><li>・ SEPTEMBER</li><li>・ OCTOBER</li><li>・ NOVEMBER</li><li>・ DECEMBER</li></ul>

### 15.3.15 monthNameL

ロケール l を使用して、日付 a またはタイムスタンプ m の月要素を表す文字列を返します。

ロケール l には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。

構文	<pre>string monthNameL(date a, string l) string monthNameL(timestamp m, string l)</pre>
例	<pre>monthNameL(toTimestamp('2001-12-30 10:12:32.222'), 'en_US') = DECEMBER monthNameL(toTimestamp('2001-12-30 10:12:32.222'), 'fr_FR') = DÉCEMBRE monthNameL(toTimestamp('2001-12-30 10:12:32.222'), 'de_DE') = DEZEMBER</pre>
制限	ラテンベースのスクリプトでは、名前が大文字で返されます。

#### 関連項目

- ・ 242 ページの [monthName](#)
- ・ 282 ページの [複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)

### 15.3.16 now

戻り値: 現在のタイムスタンプ

date

日付と時間を表すタイムスタンプの値を返します。この関数はシステム関数で、以下の特徴があります。

- ・ 非決定的です。
- ・ データソースからではなく、データフェデレーションサービスからの値を返します。

### 15.3.17 quarter

日付 *a* またはタイムスタンプ *m* が属する四半期を示す整数 1 ～ 4 を返します。値 1 は 1 月 1 日～ 3 月 31 日の範囲を示します。

構文	<code>integer quarter(date a)</code> <code>integer quarter(timestamp m)</code>
----	---

### 15.3.18 second

時刻 *t* またはタイムスタンプ *m* の秒のコンポーネントを示す整数 0 ～ 59 を返します。

構文	<code>integer second(time t)</code> <code>integer second(timestamp m)</code>
----	---

### 15.3.19 timestampadd

タイムスタンプ *m* に間隔の *count* 数を加えたタイムスタンプを返します。

構文	<pre>timestamp timestampadd(string interval-constant, integer count, timestamp t)</pre> <pre>timestamp timestampadd(integer interval-constant, integer count, timestamp t)</pre> <p>間隔の定数として、以下のいずれかの値を使用できます。</p> <ul style="list-style-type: none"><li>・ 'SQL_TSI_FRAC_SECOND' (0)</li><li>・ 'SQL_TSI_SECOND' (1)</li><li>・ 'SQL_TSI_MINUTE' (2)</li><li>・ 'SQL_TSI_HOUR' (3)</li><li>・ 'SQL_TSI_DAY' (4)</li><li>・ 'SQL_TSI_WEEK' (5)</li><li>・ 'SQL_TSI_MONTH' (6)</li><li>・ 'SQL_TSI_QUARTER' (7)</li><li>・ 'SQL_TSI_YEAR' (8)</li></ul>
制約事項	<ul style="list-style-type: none"><li>・ 演算結果は、'SQL_TSI_HOUR' の地域の夏時間によって異なります。</li></ul>

### 15.3.20 timestampdiff

timestamp 2 と timestamp 1 との差 (timestamp 2 > timestamp 1) を示す間隔数を整数で返します。

構文	<pre>integer timestampdiff(string interval-constant, timestamp m1, timestamp m2)  integer timestampdiff(integer interval-constant, timestamp m1, timestamp m2)</pre> <p>間隔の定数として、以下のいずれかの値を使用できます。</p> <ul style="list-style-type: none"><li>・ 'SQL_TSI_FRAC_SECOND' (0)</li><li>・ 'SQL_TSI_SECOND' (1)</li><li>・ 'SQL_TSI_MINUTE' (2)</li><li>・ 'SQL_TSI_HOUR' (3)</li><li>・ 'SQL_TSI_DAY' (4)</li><li>・ 'SQL_TSI_WEEK' (5)</li><li>・ 'SQL_TSI_MONTH' (6)</li><li>・ 'SQL_TSI_QUARTER' (7)</li><li>・ 'SQL_TSI_YEAR' (8)</li></ul>
制約事項	<ul style="list-style-type: none"><li>・ 差が非常に大きい場合、例外が発生することがあります。</li><li>・ 現在、演算結果は SQL_TSI_HOUR を示す地域の夏時間によって異なります。</li><li>・ 週の最初の日は日曜です。</li></ul>

### 15.3.21 trunc

タイムスタンプ m の日付より下の部分を切り捨てて日付に丸めます。

構文	<pre>timestamp trunc(timestamp "m")</pre>
----	---

### 15.3.22 week

日付 *a* またはタイムスタンプ *m* の、週番号を表す整数 1 ～ 53 を返します。1 週間は、日曜で始まり土曜で終わります。

構文	<code>integer week(date a)</code> <code>integer week(timestamp m)</code>
制約事項	週の最初の日は日曜です。第 1 週には少なくとも 1 日が含まれます。  1 月 1 日が土曜日の場合、次の規則が適用されます。 <ul style="list-style-type: none"><li>・ 1 月 1 日は第 1 週と見なされます。</li><li>・ 1 月 2 日～ 8 日は第 2 週と見なされます。</li><li>・ 12 月 25 日～ 31 日は第 53 週と見なされます。</li></ul>

### 15.3.23 weekL

日付 *a* またはタイムスタンプ *m* の、週番号を表す整数 1 ～ 53 を返します。week は 7 日間であると定義されており、開始と終了はロケール *l* に依存します。

ロケール *l* には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。

構文	<code>integer weekL(date a, string l)</code> <code>integer weekL(timestamp m, string l)</code>
----	---

例	<pre>weekL(toTimestamp('2001-12-30 10:12:32.222'), 'en_US') = 1  weekL(toTimestamp('2001-12-30 10:12:32.222'), 'fr_FR') = 52  weekL(toTimestamp('2001-12-30 10:12:32.222'), 'de_DE') = 52  weekL(toTimestamp('2001-12-30 10:12:32.222'), 'ja_JP') = 1</pre>
制限	<p>週の初めの曜日、および年の初めの週の最低日数は、ロケール 1 に依存します。</p> <p>詳細については、Java API の Calendar クラス定義を参照してください。</p>

## 関連項目

- ・ 247 ページの [week](#)
- ・ 282 ページの [複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)
- ・ <http://java.sun.com/j2se/1.5.0/docs/api/java/util/Calendar.html>

### 15.3.24 year

日付 a またはタイムスタンプ m の年のコンポーネントを示す整数を返します。

構文	<pre>integer year(date a)  integer year(timestamp m)</pre>
----	--

## 15.4 文字列関数



### 15.4.1 ascii

文字列 *s* の左端にある文字のコード値を示す整数を返します。文字列が NULL の場合は NULL を返します。

構文	<code>integer ascii(string s)</code>
制約事項	<i>s</i> = ''(NULL 文字列) の場合、NULL を返します。

### 15.4.2 char

INTEGER *n* のアスキー値を返します。*n* は 0 以上 255 以下です。*n* が範囲外の場合は、NULL を返します。

INTEGER *n* のアスキー値を返します。*n* は 0 以上 255 以下です。*n* が範囲外の場合は、NULL を返します。

構文	<code>string char(integer n)</code>
制約事項	<i>n</i> < 0 または <i>n</i> > 255 の場合、NULL を返します。

### 15.4.3 concat

2 つの文字列を連結します。

構文	<code>string concat(string s1, string s2)</code>
例	<code>concat('AB', 'CD') = 'ABCD'</code>
制約事項	s1 = NULL または s2 = NULL の場合、NULL を返します。

#### 15.4.4 containsOnlyDigits

文字列 s が数字だけで構成される場合は true を返します。それ以外の場合は false を返します。

構文	<code>boolean containsOnlyDigits(string s)</code>
----	---

#### 15.4.5 insert

文字列 s1 の start の位置から length の文字数を削除し、s1 の start の位置に s2 を挿入することによって作られた文字列を返します。start の値は、1 から文字列 s1 の長さに 1 を加えた数までの INTEGER である必要があります。length の値は、0 から文字列 s1 の長さまでの INTEGER である必要があります。start または length の値が範囲外である場合は、NULL を返します。

構文	<code>string insert(string s1, integer start, integer length, string s2)</code>
制約事項	start の値が 1 から s1 の長さまでの範囲にない場合、または length の値が 0 より小さい場合は、NULL を返します。

## 15.4.6 isLike

マッチングパターン s2 に対する文字列 s1 を確認します。このパターンは SQL 92 標準に従います。文字列 s3 は、パターンにおけるエスケープ文字の指定に使用することができます。

'\_' または '%' が文字列 s1 にある場合、文字 s3 を定義することによって一致させることができます。 '\_' または '%' が文字列 s2 にある場合、s3 で '\_' または '%' を前に置くことによって一致させることができます。

パターンは、以下のようになります。

- 文字は次の 2 つがあります。
  - メタキャラクタ '%' (パーセント) または '\_' (アンダースコア)
  - 正規キャラクタ (メタキャラクタ以外のあらゆる文字)
- '\_' はすべての単一文字に一致します。
- '%' はすべての文字列に一致します。
- パターン s2 のあらゆる正規キャラクタは s1 の同じ文字に一致します。
- '\_' または '%' が文字列 s1 にある場合、文字 s3 を定義することによって一致させることができます。 '\_' または '%' が文字列 s2 にある場合、s3 で '\_' または '%' を前に置くことによって一致させることができます。

構文	<pre>boolean isLike(string s1, string s2)</pre> <pre>boolean isLike(string s1, string s2, string s3)</pre> <p><b>注</b> 上述の 3 番目の引数は、メタキャラクタのエスケープに使用される文字です。以下の制約事項を参照してください。</p>
例	<pre>isLike("ABCD", "AB%") = true</pre> <pre>isLike("ABCD", "AB_D") = true</pre> <pre>isLike("10000", "100%") = true</pre> <pre>isLike("10000", "100¥%", "¥") = false</pre> <pre>isLike("status: 100%", "100¥%", "¥") = true</pre>

制約事項	<ul style="list-style-type: none"><li>・ (strings1, strings2): s1 == NULL または s2 == Null の場合、NULL を返します。</li><li>・ (strings1, strings2, strings3): s1 == NULL または s2 == NULL または s3 == NULL の場合、NULL を返します。</li><li>・ (strings1, strings2, strings3): s3 では、s2 に続く文字は '_'、'%」、2 番目の s3 のいずれかである必要があります。</li></ul>
------	--

### 15.4.7 left

文字列 s の左から n 文字を返します。

構文	<code>string left(string s, integer n)</code>
エイリアス	<code>leftStr()</code>
制約事項	n <= 0 の場合は NULL を返します。

### 15.4.8 leftStr

文字列 s の左から n 文字を返します。

構文	<code>string leftStr(string s, integer n)</code>
エイリアス	<code>left()</code>
制約事項	n <= 0 の場合は NULL を返します。

### 15.4.9 len

文字列 *s* の長さを返しますスペースもカウントされます。

構文	<code>integer len(string s)</code>
エイリアス	<code>length()</code>

### 15.4.10 lPad

別の文字列 *s2* を使用して文字列 *s1* を指定された長さ *n* まで左側に埋め込みます。

構文	<code>string lPad(string s1, string s2, integer n)</code>
例	<code>lPad('AB','x', 4) = 'xxab'</code> <code>lPad('ABC','x', 2) = 'AB'</code> <code>lPad('ABC','cd', 7) = 'cdcdABC'</code>
制約事項	<i>n</i> が <i>s1</i> の長さよりも小さい場合、 <code>leftStr(s1, n)</code> を返します。 <i>n</i> ≤ 0 の場合は NULL を返します。 <i>s2</i> = ''(NULL 文字列) の場合、NULL を返します。

#### 注

*n* が *s1* の長さよりも小さい場合、*s1* を切り捨てます。

### 15.4.11 lTrim

文字列 *s* の左側から最初の連続したスペースとタブを削除します。

*s1* および *s2* を指定すると、lTrim が *s1* の左側から最初の連続した *s2* を削除します。文字列 *s2* は単一文字である必要があります。

構文	<pre>string lTrim(string s)  string lTrim(string s1, string s2)</pre>
例	<pre>lTrim(' ABCD') = 'ABCD'  lTrim(' AB CD ') = 'AB CD '</pre>
制約事項	<ul style="list-style-type: none"> <li>・ (文字列 <i>s</i>): 削除される文字は ' ', '␣', '␣' です。</li> <li>・ (string <i>s</i>): ltrim(<i>s</i>) = '' の場合、NULL を返します。</li> <li>・ (string <i>s1</i>, string <i>s2</i>): ltrim(<i>s1</i>, <i>s2</i>) = '' の場合、NULL を返します。</li> <li>・ (string <i>s1</i>, string <i>s2</i>): <i>s2</i> は単一文字である必要があります。</li> </ul>

### 15.4.12 permute

2 つのテンプレートを使用する文字列の順序を変更します。

2 番目の引数 *reference-pattern* で参考パターンが提供されている最初の文字列 *s1* から、新しいパターン *new-pattern* を適用して文字列を作成します。新しいパターンは、参考パターンで定義された文字の順序を変更することによって表されます。

- ・ 参考パターンは、文字列 *s* の各文字を *reference-pattern* の対応する位置にある文字に割り当てます。*reference-pattern* の長さは、*s* の長さと同しくする必要があります。
- ・ 新しいパターンは、参考パターンに割り当てられた文字の順序を変更します。

たとえば、文字列 *s* が日付を示す '22/09/1999' であるとき、以下のようにして '1999-09-22' に変換します。

参考パターンは DD/MM/YYYY という形です。D は日、M は月、Y は年です。文字はそれぞれの位置に従って当てはめられています。

この例では、最初の D が文字列 *s* の最初の文字、2 番目の D が *s* の 2 番目の文字、'/' が *s* の 3 番目の文字、最初の M が 4 番目の文字、となります。このため、reference-pattern の長さは常に文字列 *s* の長さと等しくする必要があります。2 つの文字列の長さが異なる場合、この関数はエラーを返します。

文字マッピングが定義されると、new-pattern によって文字列 *s* を変換する必要があります。たとえば、YYYY-MM-DD が新パターンの場合、この関数は *s* から新しい日付フォーマットへの変換を定義します。その結果、'22/09/1999' という *s* から '1999-09-22' が得られます。

文字列 <i>s</i>	22/09/1999
reference-pattern	MM/DD/YYYY
new-pattern	YYYY-MM-DD
結果	1999-22-09

参考パターンで既に使用されている文字がなければ、新しいパターンにテキストを挿入することもできます。たとえば、新パターン MM/DD Year :YYYY で 09/22 Year: 1999 という文字列が作成されます。順序変更関数は、フォーマット (日付、時刻、エンコード) の変換だけでなく、規定の長さのコードから情報を抽出する場合にも有用です (以下の例を参照してください)。

構文	<code>string permute(string s1, string reference-pattern, string new-pattern)</code>
例	<ul style="list-style-type: none"><li>日付を表現するフォーマットの変更 <code>permute('02/09/2003', 'DD/MM/YYYY', 'YYYY-MM-DD') = '2003-09-02'</code> <code>permute('02-09/200', 'DD/MM/YYYY', 'YYYY-MM-DD') = '2003-09-02'</code> <code>permute('02/09_2003', 'DD/MM/YYYY', 'DL :MM/DD An :YYYY') = 'DL :09/02 An :2003'</code></li><li>ある日付を表す文字列から月と年を抽出 <code>permute('2003-09-02', 'DDYY-MM-YY', 'MM/YY') = '09/03'</code></li><li>内部コードから番号を形成 <code>permute('03/03/21-0123', 'bbbYY/MM/DD-NNNN', 'YYM MDDNNNN') = '0303210123'</code></li><li>内部コードから日付情報を抽出 <code>permute('2003NL987M08J21', 'YYYYXXXXXXMMXDD', 'YYYY-MM-DD') = '2003-08-21'</code></li></ul>

### 15.4.13 pos

文字列 s2 において文字列 s1 が最初に現れる位置を返します。文字列 s1 が見つからない場合は 0 を返します。最初の文字の位置を 1 の位置とします。start が指定されている場合は、s2 の start の位置から検索を始めます。



構文	<code>integer pos(string s1, string s2, integer start)</code>
エイリアス	<code>locate()</code>
例	<code>pos('cd', 'abcd') = 3</code> <code>pos('abc', 'abcd') = 1</code> <code>pos('cd', 'abcdcd') = 3</code> <code>pos('cd', 'abcdcd', 3) = 3</code> <code>pos('cd', 'abcdcd', 4) = 5</code> <code>pos('ef', 'abcd') = 0</code>
制約事項	<ul style="list-style-type: none"><li>・ <code>start &lt; 1</code> の場合、<code>start = 1</code> とみなします。</li><li>・ <code>start &gt; s1</code> の長さの場合、<code>0</code> を返します。</li></ul>

#### 15.4.14 repeat

文字列 `s` の繰り返しによって形成される文字列を返します。文字列は `n` 回繰り返されます。繰り返し回数 `n` が負の値のときは `NULL` を返します。

構文	<code>string repeat(string s, integer n)</code>
制約事項	<code>n &lt;= 0</code> の場合は <code>NULL</code> を返します。

#### 15.4.15 replace

文字列 `s1` における文字列 `s2` をすべて文字列 `s3` に置き換えます。

構文	<code>string replace(string s1, string s2, string s3)</code>
例	<code>replace('rar', 'a', 'ada')</code> は、 <code>'radar'</code> を返します。
制約事項	<code>s2 = ''</code> (NULL 文字列) の場合、 <code>s1</code> を返します。 <code>s3 = ''</code> (NULL 文字列) の場合、NULL を返しません。

#### 15.4.16 `replaceStringExp`

Java 正規表現の構文に従って、文字列 `s1` における文字列 `s2` をすべて文字列 `s3` に置き換えます。

Java 正規表現に関する詳細は、<http://java.sun.com/j2se/1.4.2/docs/api/java/util/regex/Pattern.html> で Sun Java の説明文書を参照してください。

構文	<code>string replaceStringExp(string s1, string s2, string s3)</code>
----	---

#### 15.4.17 `right`

文字列 `s` の右から `n` 文字を返します。

構文	<code>string right(string s, integer n)</code>
エイリアス	<code>rightStr()</code>
制約事項	<code>n &lt;= 0</code> の場合は NULL を返します。

### 15.4.18 rightStr

文字列 *s* の右から *n* 文字を返します。

構文	<code>string rightStr(string s, integer n)</code>
エイリアス	<code>right()</code>
制約事項	<i>n</i> <= 0 の場合は NULL を返します。

### 15.4.19 rPad

別の文字列 *s2* を使用して文字列 *s1* を指定された長さ *n* まで右側に埋め込みます。

構文	<code>string rPad(string s1, string s2, integer n)</code>
制約事項	<i>n</i> が <i>s1</i> の長さよりも小さい場合、 <code>leftStr(s1, n)</code> を返します。 <i>n</i> <= 0 の場合は NULL を返します。 <i>s2</i> = ''(NULL 文字列) の場合、NULL を返します。

#### 注

*n* が *s1* の長さよりも小さい場合、*s1* を切り捨てます。

### 15.4.20 rPos

文字列 *s2* において文字列 *s1* が最後に発生した位置を返します。文字列 *s2* が見つからない場合は 0 を返します。最初の文字を位置 1 とし、カウントは左から右に行われます。

構文	<code>integer rPos(string s1, string s2)</code>
例	<code>rPos('CD', 'ABCD') = 3</code> <code>rPos('CD', 'ABCD CD') = 5</code> <code>rPos('ABC', 'ABCD') = 1</code> <code>rPos('EF', 'ABCD') = 0</code>

### 15.4.21 rTrim

文字列 `s` の右側から最初の連続したスペースとタブを削除します。

`s1` および `s2` を指定すると、`rTrim` は `s1` の右側から最初の連続した `s2` を削除します。文字列 `s2` は単一文字である必要があります。

構文	<code>string rTrim(string s)</code> <code>string rTrim(string s1, string s2)</code>
例	<code>rTrim('ABCD ') = 'ABCD'</code> <code>rTrim(' AB CD ') = ' AB CD'</code>
制約事項	<ul style="list-style-type: none"><li>・ (文字列 <code>s</code>): 削除される文字は ' ', 't', 'r' です。</li><li>・ (string <code>s</code>): <code>rtrim(s) = ''</code> の場合、NULL を返します。</li><li>・ (string <code>s1</code>, string <code>s2</code>): <code>rtrim(s1, s2) = ''</code> の場合、NULL を返します。</li><li>・ (string <code>s1</code>, string <code>s2</code>): <code>s2</code> は単一文字である必要があります。</li></ul>

### 15.4.22 space

n 個のスペースの文字列を返します。n が負の値の場合は、NULL を返します。

構文	<code>string space(integer n)</code>
制約事項	n <= 0 の場合は NULL を返します。

### 15.4.23 subString

文字列から部分文字列を返します。

この関数は、位置 n1 から始まる長さ n2 の部分文字列を文字列 s から抽出します。文字列 s が短いために n2 の長さを抽出できない場合、結果として得られる部分文字列の終わりは文字列 s の終わりに一致します。この場合、部分文字列は n2 よりも短くなります。

n2 を指定しない場合は、位置 n から s の終わりまでの部分文字列を返します。

構文	<code>string substring(string s, integer n)</code> <code>string substring(string s, integer n1, integer n2)</code>
例	<code>substring('ABCD', 2, 2) = 'BC'</code> <code>substring('ABCD', 2, 10) = 'BCD'</code> <code>substring('ABCD', 0, 2) =</code> <code>NULL</code>
制約事項	(string s, integer n): 長さが 0 以下、start が s の長さよりも大きい、start が 0 以下、s = '' の場合、NULL を返します。

### 15.4.24 toLower

文字列を小文字に変換します。

構文	<code>string toLower(string s)</code>
エイリアス	<code>lcase()</code>
例	<code>toLower('ABCD') = 'abcd'</code> <code>toLower('Cd123') = 'cd123'</code>

### 15.4.25 toUpper

文字列を大文字に変換します。

構文	<code>string toUpper(string s)</code>
エイリアス	<code>ucase()</code>
例	<code>toUpper('abcd') = 'ABCD'</code>

### 15.4.26 trim

文字列 `s` の右側および左側から最初の連続したスペースとタブを削除します。

`s1` と `s2` を指定すると、`trim` は `s1` の右側および左側から最初の連続した `s2` を削除します。文字列 `s2` は単一文字である必要があります。

構文	<pre>string trim(string s)</pre> <pre>string trim(string s1, string s2)</pre>
制約事項	<ul style="list-style-type: none"><li>・ (文字列 s): 削除される文字は ' '、'¥t'、'¥r' です。</li><li>・ (string s): trim(s) = '' の場合、NULL を返します。</li><li>・ (string s1, string s2): trim(s, s2) = '' の場合、NULL を返します。</li><li>・ (string s1, string s2): s2 は単一文字である必要があります。</li></ul>

## 15.5 システム関数

### 15.5.1 データベース

戻り値: データベースの名前

string

データベースの名前を返します (カタログ)。この関数はシステム関数で、以下の特徴があります。

- ・ 非決定的です。
- ・ データソースからではなく、データフェデレーションサービスからの値を返します。

### 15.5.2 ifElse

条件 b に基づいて値を返します。

条件 b は論理式である必要があります。

- ・ b が true となる場合、関数は 2 番目の引数を返します。
- ・ b が false となる場合、関数は 3 番目の引数を返します。

構文	<pre>boolean ifElse(boolean b, boolean b1, boolean b2) date ifElse(boolean b, date a1, date a2) decimal ifElse(boolean b, decimal c1, decimal c2) double ifElse(boolean b, double d1, double d2) integer ifElse(boolean b, integer n1, integer n2) null ifElse(boolean b, null u1, null u2) string ifElse(boolean b, string s1, string s2) timestamp ifElse(boolean b, timestamp m1, timestamp m2) time ifElse(boolean b, time t1, time t2)</pre> <ul style="list-style-type: none"><li>・ 上記のシグネチャで、3 番目の引数を null にすることができます。</li></ul>
制約事項	2 番目または 3 番目の引数のいずれかが null の場合、関数は必ずしも null を返すわけではありません。

### 15.5.3 nvl

最初の引数が Null かどうかをチェックします。

- ・ 最初の引数が Null の場合、2 番目の引数を返します。
- ・ 最初の引数が Null でない場合、その引数を返します。



構文	<code>boolean nvl(boolean b1, boolean b2)</code> <code>date nvl(date a1, date a2)</code> <code>decimal nvl(decimal c1, decimal c1)</code> <code>double nvl(double d1, double d2)</code> <code>integer nvl(integer n1, integer n2)</code> <code>string nvl(string s1, string s2)</code> <code>timestamp nvl(timestamp m1, timestamp m2)</code> <code>time nvl(time t1, time t2)</code> <code>null nvl(null u, null u)</code>
エイリアス	<code>ifNull()</code>
制約事項	引数のいずれかが Null の場合、必ずしも Null を返すわけではありません。

#### 15.5.4 user

戻り値: ユーザ名

string

ユーザ名を返します。この関数はシステム関数で、以下の特徴があります。

- ・ 非決定的です。
- ・ データソースからではなく、データフェデレーションサービスからの値を返します。

### 15.6 変換関数

### 15.6.1 cast

最初の引数 *x* を 2 番目の引数で指定された型にキャストします。

2 番目の引数はキーワードで、次の値を使用できます。

- ・ NULL
- ・ VARCHAR
- ・ DOUBLE
- ・ DECIMAL
- ・ DATE
- ・ TIME
- ・ TIMESTAMP

構文	<pre>null cast(type x AS NULL) string cast(type x AS VARCHAR) integer cast(type x AS INTEGER) double cast(type x AS DOUBLE) decimal cast(type x AS DECIMAL) date cast(type x AS DATE) time cast(type x AS TIME) timestamp cast(type x AS TIMESTAMP)</pre>
----	---

### 15.6.2 convert

最初の引数 *x* を 2 番目の引数で指定された型に変換します。

2 番目の引数は文字列の定数で、次の型の値を使用できます。

- ・ ' NULL '
- ・ ' DOUBLE '

- ・ 'DECIMAL'
- ・ 'DATE'
- ・ 'TIME'
- ・ 'TIMESTAMP'

構文	<div>null convert(type x, 'NULL')</div> <div>integer convert(type x, 'INTEGER')</div> <div>double convert(type x, 'DOUBLE')</div> <div>decimal convert(type x, 'DECIMAL')</div> <div>date convert(type x, 'DATE')</div> <div>time convert(type x, 'TIME')</div> <div>timestamp convert(type x, 'TIMESTAMP')</div>
----	---

15.6.3 hexaToInt

文字列 s の 16 進数の値を INTEGER に変換します。

構文	<div>integer hexaToInt(string s)</div>
例	<div>hexaToInt( 'AF' ) = 175</div>

15.6.4 intToHexa

INTEGER n を 16 進数の値に変換します。16 進数の値を文字列として返します。

n が 0 よりも小さい場合、この関数は  $2^{32} + n$  の 16 進数の値を返します。したがって、intToHexa(-1) = FFFFFFFF となります。

構文	<div>string intToHexa(integer n)</div>
----	--

### 15.6.5 toBoolean

引数を論理値に変換します。

- ・ 引数が文字列 *s* のとき、*s* が true、またはあらゆる大文字と小文字の組み合わせで文字列 true である場合、この関数は true を返します。それ以外の場合は、false が返されます。
- ・ 引数が論理値 *b* である場合、この関数は値 *b* を返します。
- ・ 引数が NULL である場合、この関数は NULL を返します。

構文	<code>boolean toBoolean(boolean b)</code> <code>null toBoolean(null u)</code> <code>boolean toBoolean(string s)</code>
例	<code>toBoolean('true') = true</code> <code>toBoolean('TrUe') = true</code> <code>toBoolean('tru') = false</code> <code>toBoolean('False') = false</code> <code>toBoolean('F') = false</code> <code>toBoolean('f') = false</code>
制約事項	string <i>s</i> : <code>trim(s) = ''</code> の場合、NULL を返します。

### 15.6.6 toBooleanL

ロケール *l* を使用して、引数を BOOLEAN 値に変換します。

- ・ 引数が文字列 *s* であり、*s* がロケール *l* の言語で true または文字列 true の大文字と小文字を含む変型に等しい場合、この関数は true を返します。

ロケール *l* の言語で *s* が false または文字列 false の大文字と小文字を含む変型に等しい場合には、この関数は値 false を返します。

その他の場合には、この関数はエラーを返します。

- ・ 引数が論理値 `b` である場合、この関数は値 `b` を返します。
- ・ 引数が `NULL` である場合、この関数は `NULL` を返します。

ロケール `l` には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。

構文	<pre>boolean toBooleanL(boolean b, string l) null toBooleanL(null u, string l) boolean toBooleanL(string s, string l)</pre>
例	<pre>toBooleanL('TrUe', 'en_US') = true toBooleanL('vrai', 'fr_FR') = true toBooleanL('true', 'de_DE') はエラーを返します。 toBooleanL('wahr', 'de_DE') = true toBooleanL('falsch', 'de_DE') = false</pre>
制限	<pre>string s: if trim(s) == '' は NULL を返します。</pre>

#### 関連項目

- ・ 268 ページの [toBoolean](#)
- ・ 282 ページの [複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)

### 15.6.7 toDate

文字列 `s` を日付に変換します。

文字列 `s` は `yyyy-mm-dd` と表される必要があります。`yyyy` は年、`mm` は月、`dd` は日です。

このフォーマットに従った文字列の例として、`2003-09-07` と `2003-11-29` があります。フォーマットが間違っている場合、エラーが返されます。

月、日、年の数値に制約事項はありません。月の数が 12 よりも大きい場合、または示された日の数が対応する月に存在しない場合は、`toDate` の関数が内部カレンダーを使用して正しい日付に変換します。したがって、`2003-02-29` は `'2003-03-01` に、`'2002-14-12` は `2003-02-12` に変換されます。

構文	<code>date toDate(date a)</code> <code>null toDate(null u)</code> <code>date toDate(string s)</code> <code>date toDate(timestamp m)</code>
例	<code>toDate('2003-02-12') = '2003-02-12'</code> <code>toDate('2003-02-29') = '2003-03-01'</code> <code>toDate('2002-14-12') = '2003-02-12'</code> <code>toDate('1994-110-12') = '2003-02-12'</code>

### 15.6.8 toDecimal

引数を DECIMAL に変換します。

- ・ 引数が文字列 `s` の場合、`s` は 10 進数形式である必要があり、小数点記号としてピリオドを使用します。`s` が 10 進数形式でない場合には、エラーが返されます。
- ・ 引数が 10 進、倍精度浮動小数点、整数である場合、この関数は引数の 10 進型値を返します。
- ・ 引数が NULL の場合、この関数は NULL を返します。

構文	<code>decimal toDecimal(string s)</code> <code>decimal toDecimal(decimal c)</code> <code>decimal toDecimal(double d)</code> <code>decimal toDecimal(integer n)</code> <code>decimal toDecimal(null)</code>
制約事項	(string <code>s</code> ): <code>trim(s) = ''</code> の場合、NULL を返します。

### 15.6.9 toDecimalL

ロケール l を使用して、引数を DECIMAL に変換します。

- ・ 引数が文字列 s である場合、s は 10 進数形式である必要があります。小数点記号はロケール l に依存します。s が 10 進数形式でない場合には、エラーが返されます。
- ・ 引数が 10 進、倍精度浮動小数点、整数である場合、この関数は引数の 10 進型値を返します。
- ・ 引数が NULL の場合、この関数は NULL を返します。

ロケール l には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。

構文	<code>decimal toDecimalL(string s, string l)</code> <code>decimal toDecimalL(decimal c, string l)</code> <code>decimal toDecimalL(double d, string l)</code> <code>decimal toDecimalL(integer n, string l)</code> <code>decimal toDecimalL(null, string l)</code>
例	<code>toDecimalL('1.123', 'en_US') = 1.123</code> <code>toDecimalL('1,123', 'en_US') = 1123</code> <code>toDecimalL('1 123', 'fr_FR') = 1123</code> <code>toDecimalL('1,123', 'fr_FR') = 1.123</code> <code>toDecimalL('1.123', 'de_DE') = 1123</code> <code>toDecimalL('1,123', 'de_DE') = 1.123</code>
制限	(string s): if <code>trim(s) == ''</code> は NULL を返します。

#### 関連項目

- ・ 270 ページの[toDecimal](#)」
- ・ 282 ページの[複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)」

### 15.6.10 toDouble

引数を double 値に変換します。

- ・ 引数が文字列 *s* の場合、*s* は 10 進数フォーマットである必要があり、小数部分を分ける記号としてピリオド (.) が使用されます。*s* が 10 進数形式でない場合には、エラーが返されます。
- ・ 引数が 10 進、倍精度浮動小数点、または 整数である場合、この関数はこの引数の倍精度浮動小数点型値を返します。
- ・ 引数が NULL である場合、この関数は NULL を返します。

構文	<pre>double toDouble(string s) double toDouble(decimal c) double toDouble(double d) double toDouble(integer n) double toDouble(null u)</pre>
例	<pre>toDouble ('2987.9') = 2987 toDouble ('-2987.9') = -2987.9</pre>
制約事項	(string <i>s</i> ): if trim( <i>s</i> ) == '' は NULL を返します。

### 15.6.11 toDoubleL

ロケール *l* を使用して、引数を DOUBLE に変換します。

- ・ 引数が文字列 *s* である場合、*s* は 10 進数形式である必要があります。小数点記号はロケール *l* に依存します。*s* が 10 進数形式でない場合には、エラーが返されます。
- ・ 引数が 10 進、倍精度浮動小数点、または 整数である場合、この関数はこの引数の倍精度浮動小数点型値を返します。
- ・ 引数が NULL である場合、この関数は NULL を返します。

ロケール *l* には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。



構文	<code>decimal toDoubleL(string s, string l)</code> <code>decimal toDoubleL(decimal c, string l)</code> <code>decimal toDoubleL(double d, string l)</code> <code>decimal toDoubleL(integer n, string l)</code> <code>decimal toDoubleL(null, string l)</code>
例	<code>toDoubleL('1.123', 'en_US') = 1.123</code> <code>toDoubleL('1,123', 'en_US') = 1123.0</code> <code>toDoubleL('1 123', 'fr_FR') = 1123.0</code> <code>toDoubleL('1,123', 'fr_FR') = 1.123</code> <code>toDoubleL('1.123', 'de_DE') = 1123.0</code> <code>toDoubleL('1,123', 'de_DE') = 1.123</code>
制限	(string s): if <code>trim(s) == ''</code> は NULL を返します。

## 関連項目

- ・ 272 ページの[toDouble](#)」
- ・ 282 ページの[複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)」

## 15.6.12 toInteger

引数を整数に変換します。

- ・ 引数が文字列s である場合、この関数は`floor(s)` を返します。s によって表される整数が大きすぎる場合には、エラーが返されます。
- ・ 引数が10 進、倍精度浮動小数点、または整数である場合、この関数はこの引数の整数型値を返します。
- ・ 引数が NULL である場合、この関数は NULL を返します。

構文	<code>integer toInteger(string s)</code> <code>integer toInteger(decimal c)</code> <code>integer toInteger(double d)</code> <code>integer toInteger(integer n)</code> <code>integer toInteger(null u)</code>
例	<code>toInteger ('2987') = 2987</code> <code>toInteger ('-2987') = -2987</code>
制約事項	(string s): <code>trim(s) = ''</code> の場合、NULL を返します。

### 15.6.13 toIntegerL

ロケール l を使用して、引数を整数に変換します。

- ・ 引数が文字列 s である場合、この関数は `floor(s)` を返します。s によって表される整数が大きすぎる場合には、エラーが返されます。
- ・ 引数が10 進、倍精度浮動小数点、または整数である場合、この関数はこの引数の整数型値を返します。
- ・ 引数が NULL である場合、この関数は NULL を返します。

ロケール l には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。

構文	<code>integer toIntegerL(string s, string l)</code> <code>integer toIntegerL(decimal c, string l)</code> <code>integer toIntegerL(double d, string l)</code> <code>integer toIntegerL(integer n, string l)</code> <code>integer toIntegerL(null u, string l)</code>
----	---

例	<code>toIntegerL('1.123', 'en_US') = 1</code> <code>toIntegerL('1,123', 'en_US') = 1123</code> <code>toIntegerL('1 123', 'fr_FR') = 1123</code> <code>toIntegerL('1,123', 'fr_FR') = 1</code> <code>toIntegerL('1.123', 'de_DE') = 1123</code> <code>toIntegerL('1,123', 'de_DE') = 1</code>
制約事項	(string s): if <code>trim(s) == ''</code> は NULL を返します。

## 関連項目

- ・ 273 ページの[toInteger](#)」
- ・ 282 ページの[複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)」

## 15.6.14 toNull

引数の値を NULL 値に変換します。

構文	<code>NULL toNull(BOOLEAN b)</code> <code>NULL toNull(DATE a)</code> <code>NULL toNull(DECIMAL c)</code> <code>NULL toNull(DOUBLE d)</code> <code>NULL toNull(INTEGER n)</code> <code>NULL toNull(NULL u)</code> <code>NULL toNull(String s)</code> <code>NULL toNull(TIME t)</code> <code>NULL toNull(TIMESTAMP m)</code>
----	--

## 15.6.15 toString

引数の値を STRING 値に変換します。

- ・ ある 1 つの引数を STRING に変換します。
- ・ toString(double d, integer n) や toString(decimal c, integer n) の場合、整数 n は結果の文字列に含まれる小数点以下の桁数を表します。小数は小数点以下の桁数に合わせて切り上げられます。
- ・ toString(timestamp m, string s) について、文字列 s はパターンを表します。パターンは抽出するタイムスタンプ m の要素のフォーマットを定義します。

たとえば、toString(2001-12-30 10:12:32.222, 'yyyy/MM/dd') = '2001/12/30' となります。

日付形式の詳細については、以下の URL で、Java 2 Platform API リファレンスの java.text.SimpleDateFormat クラスに関する情報を参照してください。

「<http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html>」

構文	STRING toString(BOOLEAN b) STRING toString(DATE a) STRING toString(DECIMAL c) STRING toString(DOUBLE d) STRING toString(INTEGER n) string toString(NULL u) string toString(STRING s) STRING toString(TIME t) STRING toString(TIMESTAMP m) STRING toString(DECIMAL c, INTEGER n) STRING toString(DOUBLE d, INTEGER n) STRING toString(TIMESTAMP m, STRING s)
エイリアス	str()

例	<pre>toString(45) = '45' toString(-45) = '-45' toString(45.9) = '45.9' toString(-45.9) = '-45.9' toString('2002-09-09') = '2002-09-09' toString('23:08:08') = '23:08:08' toString('2002-03-03 23:08:08.0') = '2002-03-03 23:08:08' toString(true) = 'T' toString(false) = 'F'</pre>
制約事項	<ul style="list-style-type: none"><li>・ (DOUBLE d, INTEGER n) : n は定数である必要があります。</li><li>・ (DECIMAL c, INTEGER n) : n は定数である必要があります。</li></ul>

### 15.6.16 toStringL

ロケール l を使用して、引数値を STRING 値に変換します。

- ・ ある 1 つの引数を文字列に変換します。
- ・ toStringL(double d, integer n, locale l) および toStringL(decimal c, integer n, locale l) では、integer n は結果文字列に含まれる部分桁数を表します。小数は小数点以下の桁数に合わせて切り上げられます。

日付フォーマットの詳細については、Java 2 Platform API リファレンスの java.text.SimpleDateFormat クラスを参照してください。

ロケール l には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。

構文	<pre>string toStringL(boolean b, string l) string toStringL(date a, string l) string toStringL(decimal c, string l) string toStringL(double d, string l) string toStringL(integer n, string l) string toStringL(null u, string l) string toStringL(string s, string l) string toStringL(time t, string l) string toStringL(timestamp m, string l) string toStringL(decimal c, integer n, string l) string toStringL(double d, integer n, string l)</pre>
エイリアス	str()
例	<pre>toStringL(toTimestamp('2001-12-30 10:12:32.222'), 'en_US') = 12/30/01 10:12 AM toStringL(toTimestamp('2001-12-30 10:12:32.222'), 'fr_FR') = 30/12/01 10:12 toStringL(toTimestamp('2001-12-30 10:12:32.222'), 'de_DE') = 30.12.01 10:12 toStringL(toTimestamp('2001-12-30 10:12:32.222'), 'yyyy/MM/dd', 'en_US') はエラーを返します。 toStringL(1.123, 2, 'en_US') = 1.12 toStringL(1.123, 2, 'fr_FR') = 1,12 toStringL(1.123, 2, 'de_DE') = 1,12</pre>
制限	<ul style="list-style-type: none"> <li>・ (double d, integer n, string l): n は定数である必要があります。</li> <li>・ (decimal c, integer n, string l): n は定数である必要があります。</li> </ul>

## 関連項目

- ・ 276 ページの[toString](#)]
- ・ 282 ページの[複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)]

・ <http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html>

### 15.6.17 toTime

引数を時刻に変換します。

- ・ 引数が文字列 *s* である場合、この関数は *s* を時刻に変換します。この文字列は HH:MM:SS のフォーマットで表します。HH は時、MM は分、SS は秒です。

このフォーマットに従った文字列の例として、23:09:07 と 03:11:29 があります。フォーマットが間違っている場合、エラーが返されます。時、分、秒の値に制限はありません。分または秒の数が 60 より大きい場合、または時の数が 24 より大きい場合、toTime() 関数は内部クロックを使用して正しい時間に変換します。したがって、'0:450:29' は '07:30:29' に、'25:14:180' は '01:17:00' に変換されます。

- ・ 引数が日付、時刻、タイムスタンプの場合、この関数は引数を時刻に変換します。
- ・ 引数が NULL の場合、この関数は NULL を返します。

構文	TIME toTime(String s) TIME toTime(Date a) TIME toTime(Time t) TIME toTime(TIMESTAMP m) TIME toTime(NULL u)
例	toTime('02:10:09') = '02:10:09' toTime('0:450:29') = '07:30:29'

### 15.6.18 toTimeL

ロケール *l* を使用して、引数を時刻に変換します。

- ・ 引数が文字列 *s* である場合、この関数は *s* を時刻に変換します。この文字列のフォーマットは hh:mm:ss である必要があります。hh は時、mm は分、ss は秒を表します。

このフォーマットに従った文字列の例は 23:09:07 と 03:11:29 です。フォーマットが間違っている場合、エラーが返されます。時、分、秒の値に制限はありません。分数または秒数が 60 を超えている場合、または

時間数が 24 を超えている場合には、toTime 関数が内部クロックを使用して正しい時間に変換します。そのため、0:450:29 は 07:30:29 に変換され、25:14:180 は 01:17:00 に変換されます。

- ・ 引数が日付、時刻、またはタイムスタンプである場合、この関数はこの引数を時刻に変換します。
- ・ 引数が NULL である場合、この関数は NULL を返します。

ロケール1には、複数ソースを使用するデータファンデーションの関数用のロケールコード書式を使用します。

構文	<pre>time toTimeL(string s, string l) time toTimeL(date a, string l) time toTimeL(time t, string l) time toTimeL(timestamp m, string l) null toTimeL(null u, string l)</pre>
例	<pre>toTimeL('02:10:09', 'en_US') = 02:10:09 toTimeL('0:45', 'fr_FR') = 00:45:00</pre>

#### 関連項目

- ・ 279 ページの[toTime](#)]
- ・ 282 ページの[複数のソースを使用するデータファンデーションの関数のためのロケールコード](#)]

### 15.6.19 toTimestamp

引数をタイムスタンプに変換します。

- ・ 引数が文字列 s である場合、この関数は s をタイムスタンプに変換します。この文字列 s は、YYYY-MM-DD HH:mm:ss (ssss) となります。YYYY は年、MM は月、DD は日、HH は時、mm は分、ss は秒、ssss はミリ秒です。

このフォーマットに従った文字列の例として、'2003-02-17 23:09:07' と '2003-11-12 03:11:29' があります。

- ・ Timestamp( s1, s2 ) について、文字列 s2 はパターンを表します。パターンは抽出する文字列 s1 の要素のフォーマットを定義します。

たとえば、Timestamp('4:30:26 PM on January 3, 1976', 'KK:mm:ss a ¥'on¥' MMMM d, yyyy') = 1976-01-03 16:30:26.0 となります。

DATE フォーマットの詳細については、次の URL で『Java 2 Platform API Reference for the java.text.SimpleDateFormat class』を参照してください。

<http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html>.



フォーマットが間違っている場合、エラーが返されます。年、月、日、時、分、秒の数に制限事項はありません。

月の数が 12 よりも大きい場合、または示された日の数が対応する月に存在しない場合、分もしくは秒の数が 60 より大きい場合、時の数が 24 より大きい場合、timestamp の関数は内部カレンダーを使用して正しい日付に変換します。したがって、'2002-09-09 25:14:180' は '2002-09-10 01:17:00' に変換されます。

構文	<code>TIMESTAMP toTimestamp(String s)</code> <code>TIMESTAMP toTimestamp(String s1, String s2)</code> <code>TIMESTAMP toTimestamp(Date a)</code> <code>TIMESTAMP toTimestamp(Time t)</code> <code>TIMESTAMP toTimestamp(TIMESTAMP m)</code> <code>TIMESTAMP toTimestamp(NULL u)</code>
例	<code>toTimestamp('2003-02-12 02:10:09') = '2003-02-12 02:10:09.0'</code> <code>toTimestamp('2003-02-29 02:10:09') = '2003-03-01 02:10:09.0'</code> <code>toTimestamp('2002-14-12 02:10:09') = '2003-02-12 02:10:09.0'</code> <code>toTimestamp('1994-11-12 02:10:09') = '2003-02-12 02:10:09.0'</code> <code>toTimestamp('2003-02-12 0:450:29') = '2003-02-12 07:30:29.0'</code> <code>toTimestamp('2002-09-09 25:14:180') = '2002-09-09 01:17:00.0'</code>
制約事項	(time t) : 定数の日付の値は 1970-01-01 です。

### 15.6.20 val

文字列 s を10 進型値に変換します。

文字列 s は小数フォーマットである必要があり、小数部分を分ける記号としてピリオドが使用されます。s が小数フォーマットではない場合、エラーが返されます。

構文	DECIMAL val(String s)
例	val('2987.9') = 2987.9 val('-2987.9') = -2987.9 val('UUYGV76') = 0.0
制約事項	trim(s) == '' の場合、NULL を返します。

## 15.7 複数のソースを使用するデータファンデーションの関数のためのロケールコード

SQL 式で、ロケール l を指定する必要がある場合には、以下の形式で入力する必要があります。

ロケール l は形式 ln\_CY で表示される必要があります。ln および CY は以下のように定義されています。

- ・ 言語 ln は以下の ISO コードのいずれかになります。
  - ・ ドイツ語: de
  - ・ 英語: en
  - ・ スペイン語: es
  - ・ フランス語: fr
  - ・ イタリア語: it
  - ・ 日本語: ja
- ・ 国 CY は ISO 3166-1-alpha-2 に定義されている 2 文字の国コードです。

### 関連項目

- ・ [http://www.iso.org/iso/country\\_codes/iso\\_3166\\_code\\_lists/english\\_country\\_names\\_and\\_code\\_elements.htm](http://www.iso.org/iso/country_codes/iso_3166_code_lists/english_country_names_and_code_elements.htm)

## より詳しい情報

情報リソース	場所
SAP BusinessObjects 製品情報	<a href="http://www.sap.com">http://www.sap.com</a>
SAP ヘルプ ポータル	<p><a href="http://help.sap.com/businessobjects/">http://help.sap.com/businessobjects/</a> へアクセスし、[SAP BusinessObjects Overview] サイドパネルから [All Products] をクリックします。</p> <p>SAP ヘルプ ポータルでは、すべての SAP BusinessObjects 製品とそのデプロイメントについて扱った最新のドキュメンテーションにアクセスできます。PDF 版またはインストール可能な HTML ライブラリのダウンロードが可能です。</p> <p>一部のガイドは SAP サービス マーケットプレイスに格納されており、SAP ヘルプ ポータルからは入手できません。ヘルプ ポータルのガイド一覧で、そのようなガイドには SAP サービス マーケットプレイスへのリンクが付いています。保守契約を締結されたお客様には、このサイトにアクセスするための正規ユーザー ID が付与されます。ID の入手方法については、お客様担当のカスタマー サポート担当者までお問い合わせください。</p>
SAP サービス マーケットプレイス	<p><a href="http://service.sap.com/bosap-support">http://service.sap.com/bosap-support</a> &gt; ドキュメンテーション</p> <ul style="list-style-type: none"> <li>・ インストール ガイド: <a href="https://service.sap.com/bosap-instguides">https://service.sap.com/bosap-instguides</a></li> <li>・ リリース ノート: <a href="http://service.sap.com/releasenotes">http://service.sap.com/releasenotes</a></li> </ul> <p>SAP サービス マーケットプレイスには、一部のインストール ガイド、アップグレードおよび移行ガイド、デプロイメント ガイド、リリース ノート、サポート対象プラットフォームに関するドキュメントが格納されています。保守契約を締結されたお客様には、このサイトにアクセスするための正規ユーザー ID が付与されます。ID の入手方法については、お客様担当のカスタマー サポート担当者までお問い合わせください。SAP ヘルプ ポータルから SAP サービス マーケットプレイスにリダイレクトされた場合は、左側のナビゲーション ペインのメニューを使用して、アクセスするドキュメンテーションが含まれているカテゴリを探します。</p>
Docupedia	<p><a href="https://cw.sdn.sap.com/cw/community/docupedia">https://cw.sdn.sap.com/cw/community/docupedia</a></p> <p>Docupedia は追加のドキュメンテーションリソース、協調的なオーサリング環境、および対話型のフィードバックチャネルを提供します。</p>

情報リソース	場所
開発者向けリソース	<a href="https://boc.sdn.sap.com/">https://boc.sdn.sap.com/</a> <a href="https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary">https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary</a>
SAP Community Network 上の SAP BusinessObjects に関する記事	<a href="https://www.sdn.sap.com/irj/boc/businessobjects-articles">https://www.sdn.sap.com/irj/boc/businessobjects-articles</a> これらの記事は、以前はテクニカル ペーパーという名称でした。
ノート	<a href="https://service.sap.com/notes">https://service.sap.com/notes</a> これらのノートは、以前はナレッジ ベース記事という名称でした。
SAP Community Network 上のフォーラム	<a href="https://www.sdn.sap.com/irj/scn/forums">https://www.sdn.sap.com/irj/scn/forums</a>
トレーニング	<a href="http://www.sap.com/services/education">http://www.sap.com/services/education</a> 弊社では、従来のクラス型の学習から目標を定めた eラーニング セミナーまで、学習ニーズや好みの学習スタイルに合わせたトレーニング パッケージを提供しています。
オンライン カスタマー サポート	<a href="http://service.sap.com/bosap-support">http://service.sap.com/bosap-support</a> SAP サポート ポータルには、カスタマー サポート プログラムとサービスに関する情報が含まれています。また、さまざまなテクニカル情報およびダウンロードへのリンクも用意されています。保守契約を締結されたお客様には、このサイトにアクセスするための正規ユーザー ID が付与されます。ID の入手方法については、お客様担当のカスタマー サポート担当者までお問い合わせください。
コンサルティング	<a href="http://www.sap.com/services/bysubject/businessobjectsconsulting">http://www.sap.com/services/bysubject/businessobjectsconsulting</a> コンサルタントは、初期の分析段階からデプロイメントプロジェクトの実現まで一貫したサポートを提供します。リレーショナル データベースと多次元データベース、接続、データベース設計ツール、カスタマイズされた埋め込みテクノロジーなどのトピックに関する専門的なサポートを行います。

# 索引

## 記号

.unv ユニバース  
アップグレード 37  
概要 37  
変換 37  
ローカルファイルを開く 46

## A

ANSI 92  
ユニバースパラメータ 209  
AUTO\_UPDATE\_QUERY  
ユニバースパラメータ 210

## B

BEGIN\_SQL  
ユニバースパラメータ 210  
BLOB\_COMPARISON  
ユニバースパラメータ 211  
BOUNDARY\_WEIGHT\_TABLE  
ユニバースパラメータ 212

## C

cast  
関数 266  
COLUMNS\_SORT  
ユニバースパラメータ 213  
conversion 関数 265  
CUMULATIVE\_OBJECT\_WHERE  
ユニバースパラメータ 213

## D

DISABLE\_ARRAY\_FETCH\_SIZE\_OPTIMIZATION  
ユニバースパラメータ 214  
DISTINCT\_VALUES  
ユニバースパラメータ 215

## E

END\_SQL  
ユニバースパラメータ 215  
EVAL\_WITHOUT\_PARENTHESES  
ユニバースパラメータ 216

## F

FORCE\_SORTED\_LOV  
ユニバースパラメータ 217

## I

INNERJOIN\_IN\_WHERE  
ユニバースパラメータ 217

## J

JOIN\_BY\_SQL  
ユニバースパラメータ 218

## M

MAX\_INLIST\_VALUES  
ユニバースパラメータ 219

## R

REPLACE\_COMMA\_BY\_CONCAT  
ユニバースパラメータ 219

## S

SAP NetWeaver BW  
接続パラメータ 58  
SAS  
接続パラメータ 60  
プロパティ 61  
リソース 61  
SELFJOINS\_IN\_WHERE  
ユニバースパラメータ 220  
SHORTCUT\_BEHAVIOR 220  
SMART\_AGGREGATE  
ユニバースパラメータ 221

## T

THOROUGH\_PARSE  
ユニバースパラメータ 222  
TRUST\_CARDINALITIES  
ユニバースパラメータ 222

## U

UNICODE\_STRINGS  
ユニバースパラメータ 223

## い

依存関係の表示 71

## か

### 関数

abs 228  
acos 229  
ascii 249  
asin 229  
atan 229  
atan2 229  
cast 266  
ceiling 230  
char 249  
concat 249  
containsonlydigits 250  
convert 266  
cos 230  
cot 230  
curdate 237  
curtime 237  
dayname 237  
dayNameL 238  
dayOfMonth 239  
dayOfWeek 239  
dayOfWeek 239  
dayOfYear 240  
decrementdays 240  
degrees 231  
exp 231  
floor 231  
hexaToInt 267  
hour 241  
ifElse 263  
ifNull 264  
incrementdays 241  
intToHexa 267  
lPad 253  
isLike 251  
lcase 262  
left 252  
leftStr 252  
len 253  
length 253

## 関数 (続き)

log10 232  
lTrim 254  
minute 241  
mod 232  
month 242  
monthname 242  
monthNameL 243  
now 243  
nvl 264  
permute 254  
pi 233  
pos 256  
quarter 244  
radians 233  
rand 234  
repeat 257  
replace 257  
replaceStringExp 258  
right 258, 259  
rightStr 258, 259  
round 234  
rPad 259  
rPos 259  
rTrim 260  
second 244  
sign 235  
sin 235  
space 261  
sqrt 235  
str 276  
String 277  
subString 261  
tan 236  
timestampadd 244  
timestampdiff 245  
toBoolean 268  
toDate 269  
toDecimal 270, 271, 272  
toDouble 272  
toInteger 273, 274  
toLower 262  
toNull 275  
toString 276  
toTime 279  
toTimestamp 280  
toUpper 262  
trim 262  
trunc 236, 246  
truncate 236  
ucase 262  
val 281  
week 247  
year 248  
検索 256  
システムタイプ 263

## 関数 (続き)

数値 228  
挿入 250  
データベース 263  
日付/時刻 237  
変換型 265  
文字列型 248  
ユーザ 265  
累乗 233  
ログ 232  
ロケールコード 282

## く

## クエリ

作成 148  
フィルタの追加 151  
複合 151  
プレビュー結果 141  
プロパティ 148

## クエリスクリプト

編集 137

## クエリの最適化

統計の計算

複数ソースユニバース用 134, 135

複数ソースユニバース用 134, 135

## クエリパネル

クエリ結果のプレビュー 141  
[クエリ]枠 140  
[結果オブジェクト]枠 139  
説明 137  
フィルタ枠 139

## け

結合ステータス 90

## し

時間列 84  
システム関数 263  
修飾子 78  
所有者 78  
シングルサインオン(SSO)  
MSAS 2005 65

## す

数値  
関数 228  
スクリプト  
編集 137  
ステータス  
結合 90  
テーブル 90

## せ

## 接続パラメータ

SAP NetWeaver BW 58  
SAS 60  
データソース 58, 60

## て

## データソース

接続パラメータ 58, 60

データベーステーブル 78

## テーブル

オーナー 78

修飾子 78

テーブルステータス 90

## と

ドメイン名 175

## に

## 認証

MSAS 接続 65  
SAP NetWeaver BW 接続 64

## ひ

日付/時刻

関数 237

標準テーブル 78

## 開く

ローカルに保存された .unv ユニバー  
ス 46

## ふ

## フィルタ

クエリに追加 151

## 複数ソースユニバース

クエリの最適化 134, 135

## プレビュー

クエリ結果 141

## へ

## 編集

クエリスクリプト 137

## め

## メンバーセレクト

概要 141

## も

文字列関数 248

## ゆ

ユニバース

ローカルに保存された .unv ファイル  
を開く 46

## れ

列データの分析 81

## ろ

ローカライズされた関数

dayNameL 238  
dayOfWeekL 239  
monthNameL 243

ローカライズされた関数 (続き)

StringL 277  
toBooleanL 268  
toDecimalL 271  
toDoubleL 272  
toIntegerL 274  
toTimeL 279  
weekL 247

ロケール

関数コード 282

