# Peregrine AssetCenter AssetCenterの高度な使い方



DAC-42-JP08

© Copyright 2002 Peregrine Systems, Inc.

All rights reserved.

本書に記載されている情報は、Peregrine Systems, Incorporatedが所有し、Peregrine Systems, Inc. の書面による許可なく使用または開示することはできません。本書の一部または全部を、 Peregrine Systems, Inc.の事前の書面による許可なく無断で複製することを禁じます。本書に記載されている商品名は、該当する各社の商標または登録商標です。

Peregrine Systems ®およびAssetCenter ®は、Peregrine Systems, Inc.の商標です。

本書で説明されているソフトウェアは、Peregrine Systems, Inc.とエンドユーザ間で締結される ライセンス契約に基づいて提供されます。契約の条項に従って、ソフトウェアを使用する必要 があります。Peregrine Systems, Inc.は、本書の内容については一切の責任を負いかねます。ま た、本書の内容が予告なく変更されることもあります。本書の最終バージョンの日付を確認す るには、Peregrine Systems, Inc.のカスタマサポートまでお問合せください。

デモ用データベースと本書の例に使用されている団体名および個人名は架空のものであり、本 ソフトウェアの使用方法を説明するためのものです。現在、過去を問わず、実在する団体や個 人とのいかなる類似もまったくの偶然によるものです。

本製品に関する技術情報の請求、またはライセンスをお持ちの製品に関するマニュアル類の請求については、Peregrine Systemsのカスタマサポート(support@peregrine.com)までお寄せください。

本マニュアルに関するご意見やご要望は、Peregrine Systems, Inc.の出版部 (doc\_comments@peregrine.com)までお寄せください。

本書の内容は、ライセンス契約に基づくプログラムのバージョン4.2.0に適用されます。

AssetCenter

Peregrine Systems, Inc. Worldwide Corporate Campus and Executive Briefing Center 3611 Valley Centre Drive San Diego, CA 92130 Tel 800.638.5231 or 858.481.5000 Fax 858.481.1751 www.peregrine.com



目次

1. リストデータ	19
不特定のリストデータ	19 20 20 21 21 21
2. 履歴 · · · · · · · · · · · · · · · · · ·	23
レコードを作成する............................ テーブル内のフィールドまたは1リンク(単純リンク)を変更する(例:資	24
産のユーザ)・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	24
	25
リンク(別のテーブルへのnリンク)を削除する..........	25
リンク(別のテーブルへのnリンク)を変更する..........	25
レコードに関連付けられている任意管理項目の履歴を保持する	26
	26
	26
	27
履歴頃目の作成、削除、または发更..................	27

履歴項目を作成する.........................	27
3. AQL <b>クエリ</b>	29
はじめに...........................	29
AQL	29
AssetCenterのクエリ........................	31
AQLクエリの記述に関する推奨事項	32
AQL結合について...........................	32
主キーが0のレコードの理由と有用性	33
	35
Self	36
CurrentUser	36
システムリストデータ	37
階層構造のテーブル	38
	39
がべちまとインデックス	<u>41</u>
	<u>4</u> 1
// ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	/1
インテンテスを使う方法・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	12
	12
	43
	43
11.11 エン・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	43
	44
	40
	47
氏を記述9る........................... 空粉	47
	40
AQLのシノタックス	48
	48
クエリのシノタックス・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	49
	50
	55
	56
	57
HAVING句	57
ORDER BY句	58
INSERT句	59
UPDATE句	60
DUPLICATE句..........................	60
DELETE句..........................	60
AQLの関数	61
集合型のAQL関数...........................	61
文字列型のAQL関数...........................	62

日付型のAQL関数............................	62
数値型のAQL関数.............................	64
テスト型のAQL関数...........................	65
クエリの例.............................	65
メインテーブルのフィールドを特定の値と比較する.......	66
メインテーブルのリンクを別のリンクと比較する	66
メインテーブルのリンクを特定の値と比較する	66
メインテーブルにリンクしたテーブルのフィールドの値に応じて比較す	
る 	66
して、 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	66
	67
2000000000000000000000000000000000000	67
	67
	67
式(レコードを快系する	67
木八刀のノイールトを快系する	07
リノクなしのレコートを快終する....................................	68
エイリアスを使ったクエリ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	69
	74
4. 音九	71
書式の定義.............................	71
既製の書式を実際に使用するデータベースにインストールする....	71
データベース作成時の書式のインポート	72
既存のデータベース内に書式をインポートする	72
書式を作成する	72
	73
■ 土 イロンは 一 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	73
目からり ジェント CMM ペンジェント トート・トート トート・トート 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日	74
	74
	75
リスト・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	75
一回隊・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	75
盲式内のオフクエクトのフロハフィ · · · · · · · · · · · · · · · · · · ·	75
	70
ノロハナイ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	70
クリットを使つ....................................	78
青式のヘーン設定・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	78
正期レハートを間単に作成9 る力法	79
センユールに関連9る書式の識別	80
5. Crystal Reports	81
レポート作成プログラムの操作とインストール	81
	81
Crystal Reports完全版またはランタイムのインストール	83

既製のCrystal Reportsレポートを実際に使用するデータベースにインス	
	83
	84
	85
Seagate Crystal Reportsのレポートを変更する..............	85
Crystal Reports統計..........................	86
属性:::::::::::::::::::::::::::::	86
[自動更新]ボタン	86
<b>[ズーム]</b> ボタン	86
詳細レポートを作成する.........................	86
利用例..............................	87
Crystal Reportsでレポートを設定する...................	87
レポートを印刷する...........................	88
モジュールに関連するCrystalレポートの識別............	89
6. <b>アクション</b>	91
アクションの定義..........................	91
機能ドメイン..............................	92
アクションを作成する.........................	92
アクションのタイプ.......................	93
一般的な作成方法............................	96
<b>[DDE]タブページに</b> 入力する.....................	97
<b>[メッセージ]</b> タブページに入力する..............	99
アクションの例..........................	101
<b>[実行可能ファイル]</b> タイプのアクション例 ..........	101
[DDE]タイプのアクション例.................	102
[ <b>メッセージ</b> ]タイプのアクション例............	106
「 <b>スクリプト</b> 」タイプのアクション例............	106
変数を使う...............................	108
Svbase SQL Anvwhereデータベースエンジンを使う場合	109
アクションをテストする	109
	109
	109
アクションを実行する	109
リストで複数のレコードを選択	110
	110
7. <b>メッセージ</b>	111
メッセージシステムの仕組み.......................	111
メッセージを送信する方法.......................	112
メッセージを参照する方法	112
配信済通知..............................	113

<sup>6 ►</sup> AssetCenter 4.2.0 - AssetCenterの高度な使い方

AssetCenterの高度な使い方

8. <b>ワークフロー</b>	115
定義	115
ワークフロー	115
ワークフローのアクティビティ	116
ワークフローのイベント......................	117
ワークフローのトランジション	117
ワークフローのタスク・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	117
ワークフローのアクティビティの担当者	118
ワークフローの実行グループの定義・・・・・・・・・・・・・・	118
概要	118
ワークフローに関連するメインテーブル・・・・・・・・・・・・・	119
ワークフローのグラフィカルエディタの使用方法	121
	121
イベント	122
トランジション	123
その他の機能	123
ワークフローの活用法	123
ワークフローを使った依頼承認手続き例	124
	124
必要条件	125
アクティビティを作成する	132
アクティビティと同時に作成されるイベントのパラメータ	137
開始イベントを作成する	138
トランジションを作成する	139
	139
ワークフローのコンテキスト	142
ワークフローのコンテキストを定義する	142
ワークフローが参昭するオブジェクト	142
特定のオブジェクトに対して進行中のワークフローを制限する	143
ワークフローの役割	144
ワークフローの役割のタイプ	144
アクティビティの担当者を定義する	146
	146
「質問」タイプのアクティビティ	146
「コーザのアクション」タイプのアクティビティ	147
「白動アクション」タイプのアクティビティ	148
「テスト/スクリプト」タイプのアクティビティ	149
「開始」アクティビティ	150
アクティビティテンプレート	150
アクティビティをトリガする	151
927	151
タスクを作成する	151
クスノミニアル シー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	152
$\prod_{i=1} j \neq j \neq i \neq i \neq j \neq i \neq j \neq i \neq j \neq j \neq$	102

進行中のダスクのリストを表示する....................................	152
ユーリダスクを夫1190	100
ユーザにダスクを割り目しる....................................	153
リークノローのダスクを官理する..................	154
$1 \land \forall F \dots \dots$	154
システムイベント	154
$\mathcal{P} = \mathcal{P} = $	155
ユーザイベント・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	155
起動の一般的な条件...........................	157
イベントの処理..........................	158
応用:同期ワークフローチャートを実行する...........	162
終了イベント...........................	163
ワークフローのトランジション	164
ワークフローのアラームと制限時間..................	164
制限時間..............................	165
ワークフローのアラーム.....................	165
ワークフローの実行グループ	166
ワークフローのトラッキング・・・・・・・・・・・・・・・・・・	167
終了済みワークフローインスタンスの削除	167
終了したワークフローインスタンスを削除する理由	167
終了済みワークフローインスタンスの自動削除	167
9. <b>データの出力および</b> SQL <b>ビューの作成</b>	173
9. <b>データの出力および</b> SQL <b>ビューの作成</b>	173 173
9. <b>データの出力および</b> SQL <b>ビューの作成</b>	173 173 173
9. <b>データの出力および</b> SQL <b>ビューの作成</b> 定義	173 173 173 174
9. データの出力およびSQLビューの作成	173 173 173 174 174
9. データの出力およびSQLビューの作成	173 173 173 174 174 174
<ol> <li>データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>出力スクリプトを使ってデータを出力する</li> <li>エーニーン</li> <li>エーキ使ってデータを出力する</li> </ol>	173 173 173 174 174 174 174
<ol> <li>データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> </ol>	173 173 173 174 174 174 174 175
<ol> <li>データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>推将事項</li> </ol>	173 173 173 174 174 174 175 175
<ol> <li>データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>批カスクリプトを定該する</li> </ol>	173 173 173 174 174 174 175 175 176
<ol> <li>データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力スクリプト</li> <li>ムット</li> <li>ムシリクエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>エー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・</li></ol>	173 173 173 174 174 174 175 175 175 176 176
<ol> <li>データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>出力スクリプトを使ってデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトの作成方法</li> </ol>	173 173 173 174 174 174 175 175 176 176 177
<ol> <li>データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>山力スクリプトを使ってデータを出力する</li> <li>「リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトの作成方法</li> <li>出力クエリを定義する</li> </ol>	173 173 173 174 174 174 175 175 176 176 177 177
<ul> <li>9. データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>エリスクリプトを定義する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトの作成方法</li> <li>出力スクリプトの出力書式</li> </ul>	173 173 173 174 174 174 175 175 176 176 177 177
<ul> <li>9. データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>エリスクリプトを使ってデータを出力する</li> <li>エリスクリプトを使ってデータを出力する</li> <li>エリスクリプトを使ってデータを出力する</li> <li>エリスクリプトを使ってデータを出力する</li> <li>エリスクリプトの作成方法</li> <li>出力スクリプトの出力書式</li> <li>SQLビューに関するアクション</li> </ul>	173 173 173 174 174 174 175 175 176 176 176 177 177 179 181
<ul> <li>9. データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>「リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトの作成方法</li> <li>出力スクリプトの出力書式</li> <li>SQLビューに関するアクション</li> <li>ニューン</li> </ul>	173 173 173 174 174 174 174 175 175 176 176 177 177 179 181 181
<ul> <li>9. データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>「リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトの作成方法</li> <li>出力スクリプトの出力書式</li> <li>SQLビューに関するアクション</li> <li>出力スクリプトを実行する</li> <li>AssetCenter Exportから出力スクリプトを実行する</li> </ul>	173 173 173 174 174 174 174 175 175 176 176 177 177 177 181 181 182
<ol> <li>データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>「リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>批力スクリプトを定義する</li> <li>出力スクリプトの作成方法</li> <li>出力スクリプトの出力書式</li> <li>SQLビューに関するアクション</li> <li>出力スクリプトを実行する</li> <li>AssetCenter Exportから出力スクリプトを実行する</li> <li>DOSから出力スクリプトを実行する</li> </ol>	173 173 173 174 174 174 175 175 176 176 177 177 179 181 181 182 182
<ol> <li>データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>[リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトの作成方法</li> <li>出力スクリプトの出力書式</li> <li>SQLビューに関するアクション</li> <li>出力スクリプトを実行する</li> <li>AssetCenter Exportから出力スクリプトを実行する</li> <li>10. スクリプト</li> </ol>	173 173 173 174 174 174 175 175 176 176 176 177 179 181 181 182 182
<ul> <li>9. データの出力およびSQLビューの作成</li> <li>定義</li> <li>出力スクリプト</li> <li>出力クエリ</li> <li>AssetCenterデータベースからデータを出力する</li> <li>リストの出力]状況依存メニューを使ってデータを出力する</li> <li>[リストの出力]状況依存メニューを使ってデータを出力する</li> <li>AssetCenterデータベースでSQLビューを管理する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトを定義する</li> <li>出力スクリプトの作成方法</li> <li>出力スクリプトの出力書式</li> <li>SQLビューに関するアクション</li> <li>出力スクリプトを実行する</li> <li>AssetCenter Exportから出力スクリプトを実行する</li> <li>ロOSから出力スクリプトを実行する</li> <li>10. スクリプト</li> </ul>	173 173 173 174 174 174 174 175 175 176 176 177 177 179 181 181 182 182 185

<sup>8 ►</sup> AssetCenter 4.2.0 - AssetCenterの高度な使い方

本バージョンのBasicに関する情報	186
データアクセス表記..........................	186
スクリプトの適用..........................	187
関数の概要・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	188
関数の定義・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	188
知み込み関数とプログラマブル関数	188
問題がたの意味というが、いたいないでは、このでは、「「」」のでは、「」」ので、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」」のい、「」。	100
Bocic 団数を公類する	100
	102
	192
	192
于順门:仕息官理項日「ナユートリアル」を作成9る........	193
	193
手順3:アルコリスムを分析および定義する	193
手順4:Basicスクリプトを作成する...............	194
手順5:Basicスクリプトをテストする...............	194
スクリプトライブラリ......................	195
概念..............................	195
スクリプトライブラリを作成する................	195
スクリプトライブラリ内に記録されているスクリプトを呼び出す..	196
ヒントと注意事項	197
プログラマブル関数を使うときの注意事項............	197
スクリプトで使用する「日付+時刻」定数の形式	197
スクリプトでの「Duration(時間)」の形式	198
システムリストデータの読取りと書込み	198
Current Iser仮想リンク	190
Basicマクリプトにコメントを付ける	200
	200
エノ ハリピ ノピーリカッシュ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	200
	201
人ソリノトの日的 · · · · · · · · · · · · · · · · · · ·	201
于順  アルコリスムを万竹のよい走我りる	201
于順2:Basic人クリノトを書く	202
手順3:Basicスクリフトをテストする..............	203
	203
スクリプトの目的........................	203
手順1:アルゴリズムを分析および定義する..........	203
手順2:Basicスクリプトを書く....................	204
手順3:Basicスクリプトをテストする...............	204
	205
II. <b>JV/7</b>	205
カレンダの概要..........................	205
カレンダの詳細画面...........................	205
カレンダの使用方法...........................	206
カレンダに影響される機能・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	206

カレンダの作成方法	206 206 207 207 208 211
12. <b>タイムゾーン</b>	213
タイムゾーンを管理する理由	213
	214
タイハゾーン機能を実装する	214
タイムゾーンを作成する	214
	214
	215
	215
<ted 2j奴の但・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・<="" td=""><td>210</td></ted>	210
<pre>&gt;Dayngnunno&gt;5 数0710</pre>	210
1) · · · · · · · · · · · · · · · · · · ·	218
AsseiCenter Server Cタイムソーノを官理する	221
夫仃9るナスト・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	221
- ナストの残皮 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	222
	222
テーダヘー人の作成時、、、、、、、、、、、、、、、、、、、、、、、、、、、、、、、、、、、、	222
テーダヘースへの接続時	224
インホートおよび出力時.......................	225
13. <b>特殊フィールド</b>	227
	~~~
特殊フィールドの定義・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	227
特殊フィールドの用途・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	228
特殊フィールドを作成する........................	228
特殊フィールドの概要・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	228
特殊フィールドの作成方法・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	230
特殊フィールドの使用方法	233
リストの設定で特殊フィールドを使う.............	233
テーブルのレコードをフィルタを使って検索する.......	234
特殊フィールドを参照する......................	234
14 <b>ウィザード</b>	237
	201
表記法	237
· 定義	238
Twip	238
	238
ノード	238

オブジェクト	230
オブジェクトと子オブジェクト	230
J = J = J = J = J = J = J = J = J = J =	200
	239
	240
	240
	240
	242
ウィザードに関する一般情報・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	242
一般的な構造とシンタックス・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	242
ノードのブロパティ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	243
宣言的モデル・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	244
定数をプロパティの値として定義する..............	244
プロパティを参照する..........................	244
スクリプトをプロパティの値として定義する...........	245
プロパティで適用できるメソッド..................	246
テーブルタイプのプロパティ...................	246
グローバル変数CurrentTableとCurrentSelectionを使う.......	247
ウィザードの連鎖化	248
事件	248
作成パラメータ	248
Basic 閏数	249
BOOT / ードの定義	249
ROOT / - ドのシンタックス	250
ROOT / - ドのプロパティ	250
POOT I = K M H J I = K	250
	250
	200
	200
	257
	258
	258
TRANSITION ノードのシンタックス	259
TRANSITION ノードのフロバティ	259
TRANSITIONノードの特徴......................	260
ROOTノードでトランジションを定義する理由.........	260
FINISHノードの定義........................	260
STARTノードの定義.........................	262
TIMERノードの定義.........................	263
倍長整数および文字列ノードの定義..................	264
コントロールノードの定義.......................	264
コントロールノードの一般的なシンタックス...........	265
コントロールのタイプと関連プロパティ	265
共通のプロパティ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	265
CheckBoxコントロール	270
$ComboBox \exists \mathcal{V} \vdash \Box = \mathcal{V}$	270

	Nation	-Du	Hon	~ ¬	<b>`</b> ,	F I	<b>—</b> _	_ II	ι.																		071
	int D	IDU		ᆡ		111		_,	<i>v</i> .	• •	•	•	• •	•	•	•	•	•	•	•	•	•	•		•	•	271
					·	1	· ·	•	·	·	•	•	•	·	·	•	·	·	·	·	•	·	·	·	•		212
L	aber	コン		ц-	- //	<u> </u>		· ·	•	•	•	•	•	·	·	·	·	·	·	·	•	•	•	•	•		211
P	rogre	esse	ar_	12		Ц.	-/	-	•	•	•	•	•	• •	•	•		•	•	•	•	•	•	•	•	·	277
C	omn	nanc	But	ton	Ξ.	ン			- //	<b>·</b> .	•		•	•	·	·	·	•	•	•	•	•	•	•	•		277
D	BLi	stBo	xЭ	ン		- 1	- /I	Γ.	•	•	•		•	•	•	·	•	•	•	•	•	•	•	•			278
D	BQι	leryl	Box	Ξ.	ント	~ C	] –	-ル	<b>،</b> ،	•	•		•	•	•		•	•	•	•	•	•	•	•	•		281
D	BEd	lit⊐	ント	<ul> <li>□</li> </ul>	—	ル																					284
D	)BTa	ble	コン	ト	<b>-</b> -	-)	V																				285
D	BPa	thコ	ン	トロ	1—	ル																					286
L	inkE	dit	コン	11	<b>-</b> -	-ル	<i>١</i> .																				286
Т	extB	lox =	コン	۲I	- L	- /เ																					288
С	HAF	RT =	יע	۴c	- L	- JL																					289
F		-01	Г '	シ	- - г	ı —	JL	,	-	-	-	-	-	-		-	-	-	-	-	-	-	-		-		291
T	ICK	FDI	Τ¬	5	- - Г	ヿ_	- 11		•	-			•	•	•	•		•	•	•	•	-	•	•	•		291
Ċ				٦`	ット	- 	ı_	. II.		•		• •	•	•	•	•	•	•	•	•	•	•	•	•			201
т				יים יוח	/	- 			_	н. Н.	•	•	•	•	•	•	•	•	•	•	·	•	•	•	•		202
I N			(1) (	ン ン		」/ 1 —	- 11	. Ц		v	·	•	•	•	• •	•		•	•	•	•	•	•	•	•	•	292
			וח: יוח:		°∟ ``	1_	- <i>1</i>		Ľ.	•	•	•	•	•	·	·	·	•	•	•	·	·	·	•	•		292
				יםי	<u>,</u>	1~1	ц. , г	_,	ν	 	•	•	• •	• •	•	•	•	•	•	•	• •	•		•	•	•	293
L L	AIE سر	۱۱۱۱ : ۱۴۰۱ :	/IEE ~~//	:DI -#	I		́ Г	Ч	-	V	·	·	·	•	•	•	•	•	•	·	·	·	·	·	·	·	293
ワイョ	リー ≤ W西∡		の1F -	= DX、 ーデー	19J	\+r	-+	- 7	·	•	·	·	·	•	·	•	•	·	•	•	·	·	·	·	·	·	293
于一	- 順1			人 ′ エ	ድን	ז^ת גת	⊺9 ∖//	ູ ຈຸ	ᆠ	>+.			± -		·	·	·	·	·	·	·	·	·	·	·		294
于	5順2		ノイ	ਸ-	- r - '	- U.	ノ称	南方义 三 <del></del>	力	法	e. —	正手	휹 9	ູລ		~ 4		#-			⊔		•	•	•		294
リー	=順3	<: ·	(7	ע.	ノト	~ ≓	3 3¥	ーか	100	-	(	Γ,	<b>イ</b> Τ.	τ —	- F (	(1))	「話っ	÷,	<u> </u>	4	Νd	~					
// _						5	- HF	+	느	_`_		-				021	ι <del>Π</del>		2 0		Ľ 9	ູ	•	·	•		290
		(力)	ルエ	デ	1	タ (	の位	吏月	防	う法	ξ.					•••		<u>ب</u>	2 A	يدن	. 9	ວ	•	•			300
1		(カ. タフ	ルエ エ <del>-</del>	デース	イ の	ー タ( 概	の使	,已 吏月 	10 月方 ·	」 う法 ・	τ.								2 A			ວ					300 300
フノ イ 新	ノンイ イング 所しい	イカム タフ ハノ	ルエ ェー ート	デース	イの作	タの概点		東月・日本		」						· ·			2 A			ວ		• • •		•	300 300 302
ノイ新ノ	/ング 行しに / ー	タフノドの	ルエートプロ	デ - ス - ス を コバ	ィの作テ	タ概成ィ	の要すを	東・る編											2 n			ວ		· · ·		•	300 300 302 302
ノイ新ノビ	/ンペ /ンペ /しレ /ー	オフノドの	ルエー エー プレ を	デスをバコ	ィの作テン	タ概成ィパ	の要すをイ	中 る編ル		う法にする行		້.	ן ג	びき	r T	、 、 、 、 、 、	ー ・ ・ ・	ੇ ਂ ਰ	- n			ີ	· ·	· · ·	· ·	•	300 300 302 302 302 303
クイ新ノウイ	/ンパ /ンパ /し / ー   / イ !	イカフノ のード	ルェープドに	デスをパコす	ィの作テンる	タ概成ィパー	の要すをイ設	- 吏・る編ルり		う 法 よう ま くう ま こう ま こう ま こう ま こう そう うちょう しょう うちょう うちょう うちょう うちょう うちょう うちょう	ς · · · · · · · · · · · · · · · · · · ·	້	ר ג	び	デノ	バッ	ー ・ ・ ク	ੇ ਂ ਰ	- n		· ·	ີ	· · ·	· · · ·	· ·	•	300 300 302 302 303 303
ノイ新ノウイ留	インレーオー	(カフノのード) ラノのード	ルェープドに .	デスをバコす	ィの作テンる	タ概成ィパー	の要すをイ般	」吏 る編ルり・	「「「「「「「「「「「「」」」。	う法して行用		້ ສ	' ג	びき	ー デノ	、 、 、 、 、 、	ー・ ・ ・ ・ ・	ੱਰ		· ·		ю	· · ·	· · ·	· · ·	•	300 300 302 302 303 303 303 303
ノイ新ノウイ貿径	インレーイザ問	(カフノのード)	ルェープドに	デスをパコす	ィの作テンる	タ概成ィパー	の要すをイ般	- 吏・る編ル内・・	「「「「「「「「」」」で、「」」「「」」「「」」「「」」「」」「」」「」」「」」「」」「」」「」」」「」」」「」」」	う たくに うう たく うう たいしょう しょう しょうしょう しょうしょうしょう しょうしょう しょう	ζ	້ ສ	י ג	びき	ー デノ'	رى برى	ー・ ・・・ ・・・ ・・・・ ・・・・・・・・・・・・・・・・・・・・・・	े रेव	- - - -	· · · · · · · · · · · · · · · · · · ·		Э	• • • • • • • • •	· · · · · · · · · · · · · · · · · · ·	· · · ·	•	290 300 302 302 303 303 303 303
クロボン クロション クロション クロション クロション ウロション ウェック ひょうしん うちょう ひょう ひょう ひょう ひょう ひょう ひょう ひょう ひょう ひょう ひ	イイトリア・軍等軍	(カノ シノのード - · · ·	ルェープドに・・・	デスをパコす	ィの作テンる	夕概成ィパー	の要すをイ般	- 吏・る編ル内・・・	▲方 ・ 東 ↓ ・・・	う法 すくしてい すくしょう こうまう こうしょう こうしょう しょうしょう しょうしょうしょう しょうしょう しょうしょう しょうしょうしょう しょうしょう しょう		້	ן . ג	びき	デ <i>リ</i>	、 、 、 、	ー ・ ・ ・ ・ ・	े • •	<u>ີ</u>	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	ຈີ	• • • • • • • • •	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	•	290 300 302 302 303 303 303 303 303 304
ン イ 新 ノ ピ イ 留 答 留 答	イイリア・夏客夏客	(カノ ラノのード	ルェープドに	デスをパコす	ィの作テンる	夕概成ィパー・・・	の要すをイ般・・・	古吏・る編ル内・・・・	(方) 東京学会	う法 すく覚問 いちょう こうけいしょう こうしょう しょうしょう しょうしょ しょうしょう しょうしょうしょう しょうしょうしょう しょうしょうしょう しょうしょう しょう		, 	ן . גר גר	びき	ー デノ'	、 、 、	・ ・ ・ ・ ・	े रें के	<u>-</u>	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	ິດ 	· · · · · · · · · · · · · · · · · · ·	• • • • • • • • •	· · · ·	•	290 300 302 302 303 303 303 303 303 304 304
クロボン クリンク うちょう うちょう うちょう ひょう ひょう ひょう ひょう ひょう ひょう ひょう ひょう ひょう ひ	イイリア・電路電路電子シレーィザ問・問・問	(カフノのード)	ルェープドに・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	デスをパコす	ィの作テンる	タ概成ィパー・・・	の要すをイ股・・・・	す吏・る編ル内・・・・・		うう すくぼう うちょう うちょう うちょう うちょう うちょう しょうしょう しょうしょうしょう しょうしょう しょうちょう しょうしょう しょう		, 	ן . ג	びき	デノ'	、 、 、	ロー・ ・・・ ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	<u>ू</u> , , ,		· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	<b>.</b>	• • • • • • • • • •	• • • • • • • • • •	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · ·	290 300 302 302 303 303 303 303 303 304 304 304
クリング クロション クリング うちょう うちょう うちょう うちょう うちょう うちょう うちょう うちょ	、 イ 所 ノ フ ・	(カフノのード)・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	ルェープドに	デスをパコす	ィの作テンる・・・・	タ概成ィパー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	の要すをイ般	t吏.る編ルり......」月 りょう	4方 「「「「「「」」」(「」)」(「」)」(「」)」(「」)」(「」)」(「」)」	うう すくぼう うちょう うちょう うちょう うちょう うちょう しょうしょう しょうしょうしょう しょうしょう ひょうしょう ひょうひょう ひょう		, 		びき	デノ'	、 、 、 、		৾৾৾৾৾৾৾৾	- - - - - - - - - - - - - - - - - - -	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	о 	• • • • • • • • • • •	••••••			230 300 302 302 303 303 303 303 303 304 304 304 304 304
> ううう うう うう うう うう うう うう うう うちょう うちょう うちょ	イイリック 「 からい した で い し た し に に に に に に に に に に に い し に い し に い に い に に に に に に に に に に に に に	ィカフノドザー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	ルェープドに・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	デスをパコす	ィの作テンる・・・・・	夕概成ィパー	の要すをイ般・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	- 吏.る編ル内....... ) 月	4方 1991 第二十十十十十十十十十十十十十十十十十十十十十十十十十十十十十十十十十十十十	うう すくぼう そうしょう こうけいしょう こうしょう こうしょう しょうしょう しょうしょうしょう しょうしょう ひょうしょう ひょう ひょうひょう ひょう ひょうひょう ひょうひょう ひょうひょう ひょうしょう ひょうひょう ひょう		・ ・ ・ ・ ・ ・ ・ ・	- · · · · · · · · · · · · · · · · · · ·	びき	デノ ・	、 、 、 、		<u>ू</u> , ,	- - - -	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	о 	••••••	••••••		• • • • • • • •	290 300 302 302 303 303 303 303 303 304 304 304 304 304
、 ウ ウ 学習会会会会会会会会会会会会会会会会会会会会会会会会会会会会会会会会会	イイリン・ 重答 重答 重答 重答 重答 重答 重答 重答 重答 重答	イカノのード・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	ルェープドに・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	デスをパコす	ィの作テンる	夕概成ィパー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	の要すをイ般・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	,吏.る編ル内.......	4方 「「「「「「「「」」」」	うう すいしょう たいしょう たいしょう たいしょう こうしょう しょうしょう ひょうしょう しょうしょう しょうしょう ひょうしょう しょうしょう ひょうしょう しょうしょう ひょうしょう ひょうしょうしょう ひょうしょう ひょうひょう ひょうしょう ひょう ひょうひょう ひょう		,	ן . 	びき	・ デノ	(ジ ( 、		৾৾৾৾		· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	<b>.</b>	••••••	••••••			290 300 302 302 303 303 303 303 303 304 304 304 304 304
> ウ ウ クロンのののです。	イイリフ・食物食物食物食物食物食物でして、食物食物で、同じ、同じ、同じ、同じ、同じ、同じ、同じ、同じ、同い同じ、同い、同じ、同い、	 カフノのード ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	ルェープドに	デスをパコす	ィの作テンる・・・・・・・	夕概成ィパー	の要すをイ般・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	t吏.る編ル内........	4方・「泉」(「泉」)・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	うった しょうしょう ひょうしょう しょうしょう しょう		・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・		びき	・ デノ'	、 、 、 、		<u></u> ק	<u>-</u>		· · · · · · · · · · · · · · · · · · ·	<b>Э</b>	••••••				290 300 302 302 303 303 303 303 303 304 304 304 304 304
2 1 年 1 年 2 年 2 年 2 年 2 年 2 年 2 年 2 年 2	イボリフ、貿易貿易貿易貿易	イタノのード・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	ルェープドに・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	デスをパコす	ィの作テンる・・・・・・・・・・・	夕概成ィパー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	の要すをイ股・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	- 東・る編ル内・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	4方 、「東」(「「「「」」)、「「」」(「」」)、「「」」(「」」)、「」」(「」」)、「」」(「」」)、「」」(「」」)、「」」、「」」、「」、「」、「」、「」、「」、「」、「」、「」、「」、「」、	うう すくぼう うちょう うちょう うちょう しょう こうしょう しょうしょう しょうしょうしょう しょうしょうしょう しょうしょう しょうしょう しょうしょうしょう しょうしょう ひょうしょう ひょうひょう ひょうひょう ひょう ひょうひょう ひょうひょうひょう ひょうひょう ひょうひょう ひょうひょ ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょうひょう ひょうひょうひょう ひょうひょう ひょうひょう ひょうひょうひょうひょうひょうひょうひょうひょうひょう ひょうひょうひょうひょうひょうひょうひょうひょうひょうひょうひょうひょうひょうひ		・ ・ ・ ・ ・ ・ ・ ・ ・ ・		びき	・ デノ'	、 、 、 、		<u>ੂ</u> • ਰ	- - -		· · · · · · · · · · · · · · · · · · ·	<b>3</b>	••••••	•••••••			290 300 300 302 303 303 303 303 303 304 304 304 304 304
ァ ウ ウ 空谷留谷留谷留谷留谷留	イイルフ・重答重答重答重答重答すクレーイが問、問、問、問、問、問、問	                                                                                                                                                                                                                                                                                                    <	ルェープドに・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	デスをパコす	ィの作テンる・・・・・・・・・・・	夕概成ィパー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	の要すをイ股・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	,吏.る編ル内.............		う 法 て そ 行 問 ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・		<b>、</b> ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・		びき	≓')'	、 、 、 、 、		<u></u> ,			· · · · · · · · · · · · · · · · · · ·	<b>9</b>					230 300 302 302 303 303 303 303 303 303
2 1 年前の1993年1993年1993年1993年1993年1993年1993年1993	イイボノフ・夏客夏客夏客夏客夏客夏客	イタノドザー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	ルェープドに・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	デスをパコす	ィの作テンる・・・・・・・・・・・	夕概成ィパー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	の要すをイ設・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	,吏.る編ル内・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	「 」 た 、 、 、 、 、 、 、 、 、 、 、 、 、 、 、 、 、 、	うう すくぼう うちょう すいりょう しょうしょう こうしょう しょうしょう しょうしょうしょう しょうしょうしょうしょう しょうしょう ひょうしょう しょうしょう しょうしょう しょうしょう しょうしょう ひょうしょう しょうしょう ひょうしょう ひょうしょう ひょうしょう ひょうしょう ひょうしょう ひょうしょう しょうしょう ひょうしょう ひょうしょう ひょうしょ ひょうしょう ひょうしょうしょう ひょうしょう ひょうしょう ひょうしょうしょう ひょうしょうしょう ひょうしょう ひょう ひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょうひょう ひょう		,		びき	ー デノ'	、 、 、 、		<u> </u>				<b>ð</b>	•••••••	••••••••			230 300 302 302 303 303 303 303 303 303
、 ウ ウ	イイボノフ・貿易貿易貿易貿易貿易貿易貿易	イタノのード - ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	ルェープドに・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	デスをパコす	ィの作テンる	夕概成ィパー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	の要すをイ般・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	す吏・る編ル内・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・		う 方 、 た 尾 間 ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・		,		びき	デノ ・ ・	で 、 、 、 、		<u>ू</u>				<b>ð</b>					230 300 302 302 303 303 303 303 303 303
、 ウウンイ 新学校 人名英格兰 人名英格兰 人名英格兰人姓氏 化乙基乙基 化乙基乙基乙基乙基乙基乙基乙基乙基乙基乙基乙基乙基乙基乙基乙基乙基	イイボノフ・貿易貿易貿易貿易貿易貿易貿易	イタノのード · · · · · · · · · · · · · · · · · · ·	ル ェープドに	デスをパコす	ィの作テンる・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	夕概成ィパ	の要すをイ般・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	₁吏.る編ル内.............」		う 方 ・ た 尾 町 ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・ ・		,		びき	・ デノ ・	、 、 、 、 、		<u>ੋ</u> * ਰ				<b>3</b>		•••••••••••			230 300 302 302 303 303 303 303

12 ► AssetCenter 4.2.0 - AssetCenterの高度な使い方

質答質答質答																											306 306 306 306 306 307
15. <b>二</b> =	1 –	・ス																									309
ニュー ニュー ニュー ニュー ニュー ニュー ニュー こ ー こ 二 二 二 二 二 二 二 二 二 二 二 二 二 二 二 二	ススいースるスペ	ひのニスのメの <b>て</b> 根根:を重ッ配 <b>の</b>	要要「表要セ信谷	・・ス示度・先業	をす ジリ員	作る ス <b>グ</b>	成 ト <b>ル</b>	ਰ -	ය 7	] (		QL	· · ·	· · · · ·	bA	IIG	rou			・・・・チ:		・ ・ ・ ・	, 7 才		・・・・・ ク	・ ・ ・ 、 ス	309 310 310 310 310 310 310 310
・・・ 【サ ニュー: ニュ	プクスマク	・ グノ を表 スマ	レー ミ示 てー	フォキ	を る 一	· 含 ・ を	<b>む</b> 起	] 動	・ (S ・	SQI 3	_名	4 :	bC	hil	IdG	Grps	S)	チ	エ・	・ ツイ	・ クガ ・		・ ック ・	ス	•	• •	311 311 311 311

図の一覧表

1.1. リストデータの詳細画面	20
3.1. クエリエディタの活用法	44
41 書式のヘッダー	79
	102
	102
	103
6.3. 参照するオフシェクトを指定するメッセーシタイフのアクションの設	
定例	106
7.1. メッセージシステムの仕組み . . . . . . . . . . .	112
8.1. ワークフローチャートの概略 . . . . . . . . . . .	116
8.2. AssetCenterのワークフロー - 概要	119
8.3. ワークフローチャートを定義できるメインテーブル	120
8.4. 実行中のワークフローに関連するメインテーブル .......	120
8.5. ワークフローチャート - 依頼の承認 . . . . . . . . . .	124
8.6. ワークフローチャート - 依頼の承認 . . . . . . . . .	139
8.7. [データベース]タイプアクションの[パラメータ]タブページ	
	156
8.8. 同期ワークフローチャートの例	162
8.9. 非同期ワークフローチャートの例	163
8.10. 終了イベント付きのワークフローチャート	164
10.1, Z / U / V / V / V / V / V / V / V / V / V	190
11.1 カレンダの詳細画面の「予定表」タブページ	207
11.1. カレングの計画画面の「プレビュー」々ゴページ	201
	211

14.1.	ウィザ-	<b>-</b> ドの構造	i .											241
14.2.	ウィザ-	-ドの編成	į.											295
14.3.	実行お。	<b>トびデバッ</b>	グ	ボ	タ:	ン								301

表の一覧表

11 シフテムリフトデータの値	21
	21
3.1. 表記法	48
3.2. 論理演算子 . . . . . . . . . . . . . . .	52
3.3. 比較演算子	53
3.4. 集合型のAQL関数	61
3.5. 文字列型のAQL 関数	62
	62
	63
	64
	04
3.9. テスト型のAQL (第200	65
8.1. ワークフローを制限する場合	158
8.2. イベントの様々な処理方法 .....................	158
8.3. イベントの様々な処理方法 . . . . . . . . . . .	160
10.1. 使用可能なデータ型	191
13.1. 特殊フィールドのタイプ . . . . . . . . . . .	228
14.1. 表記法	237
14.2. ROOTノードの論理プロパティ	250
14.3 ROOT / ードの物理プロパティ	252
	252
	200
14.5. PAGEノードの論理ノロハティ	257
14.6. PAGEノードの物理ブロバティ ................	257
14.7. PAGEノードのサブノード ....................	258

14.8. TRANSITIONノードの論理プロパティ ............	259
14.9. FINISHノードの論理プロパティ ...............	260
14.10. FINISHノードの物理プロパティ . . . . . . . . . .	261
14.11. STARTノードの論理プロパティ . . . . . . . . . .	262
14.12. TIMERノードの論理プロパティ ...............	263
14.13. 倍長整数および文字列ノードの論理プロパティ .......	264
14.14. すべてのコントロールに共通の論理プロパティ .......	265
14.15. すべてのコントロールに共通の物理プロパティ .......	267
14.16. CHECKBOXコントロールのプロパティ	270
14.17. COMBOBOXコントロールの物理プロパティ ........	270
14.18. OPTIONBUTTONSコントロールの物理プロパティ	271
14.19. LISTBOXコントロールの物理プロパティ ..........	272
14.20. LISTBOXコントロールのメソッド ..............	275
14.21. LISTBOX コントロールの必須論理プロパティ .......	276
14.22. LABELコントロールの物理プロパティ . . . . . . . . .	277
14.23. PROGRESSBARコントロールの物理プロパティ ......	277
14.24. COMMANDBUTTONコントロールの物理プロパティ	278
14.25. DBLISTBOXコントロールの物理プロパティ ........	279
14.26. DBQUERYBOXコントロールの物理プロパティ .......	281
14.27. 標準モードのDBEDITコントロールの物理プロパティ .....	284
14.28. DBPATHコントロールの必須論理プロパティ . . . . . . .	286
14.29. LINKEDITコントロールの論理プロパティ . . . . . . . .	286
14.30. TEXTBOXコントロールの物理プロパティ . . . . . . . .	288
14.31. CHARTコントロールの論理プロパティ . . . . . . . . .	289
14.32. CHARTコントロールの物理プロパティ . . . . . . . . .	289
14.33. FILEEDITLコントロールのプロパティ . . . . . . . . . .	291
14.34. TICKEDITコントロールのプロパティ ...........	291
14.35. NUMBOXコントロールのプロパティ ............	292
14.36. COMBOEDITコントロールのプロパティ ..........	293
14.37. コントロールのプロパティ ................	293
15.1. マーキーツールバーのボタン ..................	311



リストデータによって、特定のフィールドの値が標準化されるため、データ入 力が容易になります。

リストデータの値はドロップダウンリストに表示されます。リストから適切な 値を選択するだけで、フィールドに値が入ります。

AssetCenterでは、次の2種類のリストデータを管理します。

-タ

• 不特定のリストデータ

トテ

• システムリストデータ

# 不特定のリストデータ

AssetCenterの管理者は、[管理/リストデータ]メニューから不特定のリスト データにアクセスできます。

次の2種類の不特定リストデータがあります。

 ユーザ自身が作成したリストデータ。ユーザは、このリストデータを任意管 理項目にリンクできますが、フィールドにはリンクできません。フィールド へのリンクはプログラムによって自動的に割り当てられます。

AssetCenter

 データベースのフィールドにリンクしているリスト。このリストは任意管理 項目にもリンクできます。このリストデータを削除した場合やリストデータの名前を変更した場合に、このリストにリンクするはずのフィールドを含む レコードが作成されると、AssetCenterは自動的に元の名前でリストデータを 再生します(値は割り当てません)。

### リストデータの値

リストデータの詳細画面に表示される値のリストは、フィールドのドロップダ ウンリストに表示される値です。

管理者は、リストの右側に表示される■、図および■ボタンを使って値の削除、 変更、および追加を行うことができます。

#### 図 1.1. リストデータの詳細画面



# オープンリストデータ

オープンリストデータは、リストデータの詳細画面の**[タイプ]**(SQL名: seType)フィールドが**[オープン]**に設定されているリストデータです。 リストデータがオープンになっている場合、AssetCenterのユーザは、ドロップ ダウンリストに表示される以外の値をフィールドに入力できます。

ユーザが新しい値を入力すると、この値がリストデータの値の一覧に追加され ます(値の一覧はすべてのユーザで共有できます)。追加する際、作成を確認 するメッセージが表示されます。

# クローズドリストデータ

クローズドリストデータは、リストデータの詳細画面の**[タイプ]**(SQL名: seType)フィールドが**[クローズド]**に設定されているリストデータです。 クローズドリストデータの場合、AssetCenterのユーザがドロップダウンリスト に表示される値以外の値を入力することはできません。

#### 💋 注意:

リストデータの詳細画面の[**作成**] または[**変更**] ボタンをクリックすると、 管理者がリストデータに加えた変更がデータベースに保存されます。この変更 をクライアントのコンピュータで有効にするには、クライアントのコンピュー タでデータベースに接続し直す必要があります。

# システムリストデータ

システムリストデータの値のリストは、AssetCenterが定義します。管理者もユー ザもこれをカスタマイズすることはできません。

システムリストデータは、**[管理/リストデータ]**メニューでは編集できません。

### システムリストデータの値

表示される値は、データベースに保存されている値とは異なります。 データベース内では、これらの値は数値として保存されます。 以下の表では、資産の詳細画面の**[割当]**(SQL名:seAssignment)フィールド を例に取り上げています。

#### 表 1.1. システムリストデータの値

データベースに保存される値	表示される値	
0	[使用中]	
1	[在庫中]	

データベースに保存される値	表示される値	
2	[除却済]	
3	[納品待ち]	

システムリストデータを参照するには、次の方法があります。

- システムリストデータで入力したフィールドのヘルプを表示する。
- AssetCenter Database Administratorを使う。
- データベース構造が記述されている「Database.txt」ファイルを使う。



変更を記録するには、フィールドまたはリンクの履歴の保持を指定する必要が あります。以下の手順で指定します。

- 1 ポップアップメニューから[オブジェクトの設定]を選択します。
- 2 フィールドの設定画面の[全般]タブページを選択します。
- 3 [履歴保持]フィールドを[はい]に設定します。
- 4 [OK]をクリックして作成を確定します。

#### 💋 注意:

フィールドの設定画面で[OK]をクリックすると、履歴オプションに加えた変更が直ちにデータベースに保存されます。フィールドまたはリンクの履歴を保持するかどうかはAssetCenter Database Administratorでも定義できます。

フィールドまたはリンクの履歴を保持すると、AssetCenterのすべてのユーザが 履歴を利用できます。

いずれかのフィールドまたはリンクで履歴を保持することを指定すると、その テーブルのレコードの詳細画面に [履歴] タブページが表示されます。フィー

AssetCenter

ルドまたはリンクに変更を加えると、**[履歴]**タブページに変更内容の詳細を 示す履歴項目が加えられます。

履歴項目には次の情報が含まれます。

- [変更日] (SQL名: dtLastModif): 変更を行った日付
- [作成者](SQL名: Author): 変更を行ったユーザ(ログイン名と氏名)
- [フィールド] (SQL名: Field): 変更したフィールド名(ラベル)
- [元の値](SQL名: Previous Val): 変更されたフィールドの元の値(コメント型のフィールドを除く)
- [新しい値](SQL名:NewVal):変更したフィールドの新しい値(コメント型のフィールドを除く)。デフォルトでは、このフィールドはリストに表示されません。表示するには、リスト上でマウスの右ボタンをクリックして[リストの設定]メニューを選択します。

💋 注意:

旧バージョンのAssetCenterからデータベースをインポートする場合、履歴項 目の**[新しい値]**フィールドは空になります。

 [元のコメント](SQL名:memPreviousCmt):コメント型のフィールドの 修正前の値。コメント型のフィールド(最大サイズ:半角の場合で32767文 字)は、他のフィールドとデータベースに保存する方法が異なるので、処理 方法も異なります。

AssetCenterの履歴機能は、実行する操作によって以下のように異なります。

### レコードを作成する

テーブルの主キーに対応するIDフィールドの変更履歴を保持することを指定した場合は、レコードの作成が記録されます。

- 記録される内容
- [変更日]:レコードの作成日
- [作成者]:レコードの作成者
- [フィールド]:「作成」
- [**元の値]**:「作成」

# テーブル内のフィールドまたは1リンク(単純リンク) を変更する(例:資産のユーザ)

#### 記録される内容

AssetCenterの高度な使い方

- [**変更日**]: 変更した日付
- [作成者]: 変更者
- [フィールド]: 変更したフィールドの名前
- [元の値]:変更したフィールドの元の値
- [新しい値]: 変更したフィールドの新しい値

# リンク(別のテーブルへのnリンク)を追加する(例: 契約が伴う資産)

#### 記録される内容

- [変更日]: 追加した日付
- [作成者]: 追加者
- [フィールド]: リンク名
- [元の値]:リンク先レコードの参照番号
- [新しい値]: 変更したフィールドの新しい値

# リンク(別のテーブルへのnリンク)を削除する

#### 記録される内容

- [**変更日]**:削除した日付
- [作成者]: 削除者
- [フィールド]:リンク先レコードの参照番号
- [元の値]:リンク先レコードの参照番号
- [新しい値]: 変更したリンク(空)の新しい値

# リンク(別のテーブルへのnリンク)を変更する

AssetCenterでは、リンクへの変更を記録しません。変更の履歴を記録するには、 古いリンクを削除してから新しいリンクを追加する必要があります。

# レコードに関連付けられている任意管理項目の履歴を 保持する

AssetCenterでは、任意管理項目の履歴をデータベース内の他のフィールドと同様に保持できます。任意管理項目の場合は、次の操作の履歴が保持されます。

- 任意管理項目の追加
- 任意管理項目の削除
- 任意管理項目の値の変更

AssetCenterの履歴機能は、実行する操作によって以下のように異なります。

### 任意管理項目の追加

任意管理項目のパラメータの詳細画面にある[**履歴の保持**](SQL名: seKeepHistory)フィールドが[**はい**]に設定されており、[メインレコードの 作成中でも履歴を保持](SQL名:bCreationHistory)チェックボックスがオン になっている場合は、新しい任意管理項目の追加が記録されます。 記録される内容

- [変更日]:任意管理項目を追加した日付
- [作成者]:任意管理項目を追加した担当者
- [**元の値]**:「作成」
- [フィールド]:任意管理項目のSQL名

### 任意管理項目の削除

**[履歴の保持]**(SQL名: seKeepHistory)フィールドが**[はい]**に設定されて いる場合は、任意管理項目の削除が記録されます。 記録される内容

- [変更日]:任意管理項目を削除した日付
- [作成者]:任意管理項目の削除者
- **[フィールド]**: 任意管理項目のSQL名
- [元の値]:削除した任意管理項目(任意管理項目の値)
- [新しい値]:任意管理項目(空)の新しい値

### 任意管理項目の値の変更

[**履歴の保持]**(SQL名:seKeepHistory)フィールドが[**はい]**に設定されて いる場合は、任意管理項目の変更が記録されます。

記録される内容

- [**変更日**]:任意管理項目を変更した日付
- **[作成者]**:任意管理項目の変更者
- [フィールド]:任意管理項目のSQL名
- [元の値]:任意管理項目の変更前の値
- [新しい値]:変更した任意管理項目の新しい値

#### 🐓 警告:

レコードを削除すると、すべての履歴項目の削除は、同時またはAssetCenter Serverの更新時に実行されます。

### 履歴項目の作成、削除、または変更

履歴項目の作成の履歴を保持することはできません。

### 履歴項目を作成する

任意管理項目の履歴項目の作成をトリガするには、[**履歴の保持**]フィールドを[はい]に設定する必要があります。この設定を行うには、任意管理項目の 詳細画面で[パラメータ]タブを選択し、リストの右側の図ボタンをクリック します。

任意管理項目のパラメータの詳細画面が表示されます。[**履歴の保持**]フィールドは、この画面の[制限]タブページにあります。

このフィールドを[**はい**]に設定すると、AssetCenterはその任意管理項目の履 歴項目を自動的に作成します。履歴項目は、その任意管理項目に関連付けられ ているテーブルの[**履歴**]タブページに表示されます。

#### 🌻 警告:

レコードを削除すると、すべての履歴項目も同時またはAssetCenter Serverの更新 時に削除されます。履歴作成の履歴は保持できません。



本章では、AQLでクエリを書く方法について説明します。

# はじめに

本節では、AQL言語と、クエリが必要になる状況について説明します。

### AQL

AQL(Advanced Query Language)は、AssetCenterがAssetCenterデータベースに アクセスするのに使うクエリ言語であり、SQLと同じように機能します。AQL で記述したクエリは、使用中のデータベースエンジンで使う同等のSQL言語に 自動的に変換されます。

✓ 注意: AQLを使うには、SQLとデータベースに関する十分な知識が必要です。

### AQL**言語の利点**

AssetCenterデータベースのクエリには、次の理由からSQLよりもAQLの方が適しています。

### データベースの種類に無関係

AssetCenterがサポートする様々なデータベースエンジンは、すべて異なるバー ジョンのSQLを採用しており、互換性がありません。AQLは、これらのデータ ベースエンジンに関係なく使うことができます。

別のデータベースエンジンに移行した場合でも、AQLで記述したクエリは同様 に機能します。

これは、どのデータベースエンジンを使用していても、AQLがデータベースと 同一の関数セットを使用しているためです。

例えば、AQLのSubstring関数はOracle SQLのSubstrおよびMicrosoft SQL Server SQLのSubstringに相当します。

#### 最適なSQLコードの生成

AQLは、使用するデータベースエンジンに応じて最適なSQLコードを生成します。

特にインデックス機能を使用する場合は、生成されるSQLコードが大きく異な ります。例えば、【モデルID】(SQL名:Model\_IModelId)と【完全名】(SQL 名:FullName)にインデックスを強制して、モデルの完全名を検索する場合は、 次のAQLを記述します。

SELECT FIRST\_ROWS IModelId, FullName FROM amModel

生成されるSQLコードは、使用するDBMSによって異なり、DBMSごとに最適化 されます。例えば、Oracleでは次のSQLコードが生成されます。

SELECT /\*+ FIRST\_ROWS INDEX\_ASC(M1 Model\_IModeIId) \*/ M1.IModeIId, M1 .FuIIName FROM amModel M1

Microsoft SQL ServerまたはSybase SQL Serverでは、次のSQLコードが生成されます。

SELECT M1.IModelId, M1.FullName FROM amModel M1 ORDER BY M1.IModelId IBM DB2では、次のコードが生成されます。

SELECT IModelId, FullName FROM amModel OPTIMIZE FOR 100 ROWS

### AssetCenterデータベースへのアクセスを簡略化

AQLは、リンクと結合の管理を簡略化します。そのため、AQLでクエリを記述 すると、直接SQLを使う場合に比べ、速くデータベースにアクセスできます。 さらに、AQLを使うと任意管理項目へのアクセスが簡単になり、関連付けられ ているテーブルのフィールドとして直接任意管理項目を使うことができます。 AQLによって、特殊フィールドも容易に活用できます。

### AQLの特徴(SQLと比較して)

AQLはDDL(Data Definition Langage)ステートメントをサポートしません。 AQLには、結合、任意管理項目、特殊フィールドの処理を簡単にする拡張機能 があります。

#### 🌻 警告:

SQLステートメントを使って直接AssetCenterデータベースに書き込まないでください。

### AssetCenterのクエリ

クエリを使うと、特定のテーブルまたはリンクしているテーブル内の情報の選 択基準を組み合わせることができます。

次の場合にクエリを使うことができます。

- レコードリストで使うフィルタを作成する。通常は、WHERE句を使った簡単なクエリを使います。
- ビューを定義する。
- エクスポートモジュールの出力条件を定義する。
- Crystal Reportsを使ってレポートを作成する。
- ウィザードを作成する。
- AssetCenter APIを使う場合。
- AssetCenterをDDEサーバとして使う場合。

AQL(Advanced Query Language)は、AssetCenterに組み込まれたクエリ言語で あり、AssetCenterデータベースへのアクセスのみに使うことを想定しています。 AssetCenterに組み込まれているエディタを使って、次の方法でAQLクエリを作 成できます。

- グラフィカルユーザインタフェースを使って作成
- AQLで直接クエリを記述

#### 🎐 警告:

本章後半では、AQLの使い方を分かりやすく説明するために、すべてのAQLシ ンタックスを使った記述例を紹介します。特にSELECT、WHERE、およびFROM 句が説明されています。AQLクエリのWHERE句だけを使ったクエリフィルタ や、式ビルダなどの特定の機能を使うと、より簡単にクエリを作成できます(こ れは一部の句のみが表示されるためです)。ただし、後述の例をこれらの機能 用に使うことはできません。

### AQLクエリの記述に関する推奨事項

AQLによるクエリの記述を始める前に、この節をお読みください。 この節では、主に次の点について説明します。

- AQLに固有の表記
- クエリの最適な設計に有効なAQLとAssetCenterデータベースの特異性

「AQL**のシンタックス」と「**AQL**の関数」**の節に、この節の補足説明がありま す。

#### 🌗 警告:

AQLで記述したクエリには、データベース内のフィールド、リンク、およびテー ブルのSQL名(SQLName)を使います。SQL名については、「Database.txt」ファ イルを参照してください。このファイルには、データベースの構造が記述され ており、すべてのSQL名が記載されています。

このファイルは、「[AssetCenterのインストール先フォルダ]/doc/infos」にあ ります。

### AQL結合について

#### 定義

結合とは、複数のデータテーブルを1つのクエリに統合することです。

#### AQL結合

AssetCenterのデータベースの記述では、テーブルとフィールドだけでなく、テー ブル間のリンクも定義します。これによって、AQLレベルでの結合を自動化で きます。

AQLリンクは次のように表現されます。

Link[.Link[.Field]]

AQLでは、上記のように結合を簡単に処理できるので、AssetCenterデータベー スで使用する大部分のクエリを簡単に作成できます。

例

次のAQLで記述したクエリは、モデルごとに以下のデータを返します。

- モデルのID(SQL名:IModelId)
- モデルの完全名(SQL名:FullName)
- メーカ (SQL名: amBrand) に関連するテーブルの名前 (SQL名: Name)

SELECT IModeIId, FullName, Brand.Name FROM amModel

Oracle SQLまたはMicrosoft SQL Serverで記述した同じクエリは、以下の通りです。

SELECT M1.IModeIId, M1.FullName, B2.Name FROM amModel M1, amBrand B2 W HERE M1.IBrandId=B2.IBrandId

**モデル**(SQL名:amModel)のテーブルとメーカ(SQL名:amBrand)のテーブ ル間の2つの結合は、AQLでは自動的に処理されます。AssetCenterのグラフィッ ククエリエディタを使えば、ツリー構造のリストで選択したテーブルまたはリ ンクしているテーブルのフィールドをクリックするだけで、対応するAQLコー ドを作成できます。

#### 💋 注意:

OracleとDB2以外のシステムでは、結合数は1つに限られています。

Microsoft SQL Server 7とMSSQL 2000では、「amdb.ini」ファイルを変更して、 クエリ実行に関連する問題を防ぎます。次の手順に従って、接続の詳細画面で、 「amdb.ini」ファイルを変更します。

useSQL92Join=1

### 主キーが0のレコードの理由と有用性

# 主キーが「0」(ゼロ)のレコード

AssetCenterデータモジュールには、次の特異性があります。

各テーブルの主キーと外部キーは数値(32ビットの整数)です。

- レコードにリンクしていない外部キーは、「0」(NULLではない)に設定されます。
- 各テーブルには、主キーが「0」に設定された空のレコードがあります。

#### 有効性

主キーが「0」のレコードを使うと、AとBの2つのテーブル間での外部結合を使わないクエリの結果に、テーブルB内の実際のレコードにリンクしない(リンクが存在しない)テーブルAのレコードを含めることができます。つまりこのレコードは、テーブルAのレコードの内、テーブルBの「0」主キーのレコードにリンクしているものを指します。

例

次のAQLで記述したクエリは、ポートフォリオ品目の資産タグごとに、そのユー ザおよび責任者の名前を返します。

SELECT AssetTag, User.Name, Supervisor.Name FROM amPortfolio

このクエリの結果には、ユーザや責任者に割り当てられていないポートフォリ オ品目が含まれます。データベースでは、このようなポートフォリオ品目は、 部署と従業員のテーブル内で主キーが「0」のレコードにリンクされています。

### 主キーが「0」のレコードを使う理由

ここでは、外部のSQL結合を使ったクエリではテーブルBのレコードにリンクしていないテーブルAのレコードを選択できるのに、AQLではなぜ主キーが「0」のレコードを使うのかを説明します。

一部のRDBMでは複数の外部結合を処理できないという欠点がありますが、主 キーが「0」のレコードを使えば、AQLクエリから生成されたSQLコードで外部 結合を使う必要がないため、この欠点を補うことができます。

例

次のAQLクエリは、ポートフォリオ品目ごとにその資産タグとユーザの場所の 名前を検索します。結果には、ユーザのないポートフォリオ品目と、場所のな いユーザのポートフォリオ品目が含まれます。

SELECT AssetTag, user.location.name FROM amPortfolio

生成されたSQLコードでDBMSの外部結合を使った場合は、Sybase SQL Server用 に生成されたSQLコードは次のようになります。

SELECT a.AssetTag, I.name FROM amPortfolio a, amEmpIDept e, amLocation I WHE RE a.IUserId \*= e.IEmpIDeptId AND e.ILocaId \*= I.ILocaId

このコードは複数の外部結合を次々に使うので、Sybase SQL Serverではサポートされません。

しかし、部署と従業員および場所のテーブル内に主キーが「0」のレコードがあ るので、外部結合を使う必要はありません。AssetCenterは次のように外部結合 のない、SQLコードを生成します。

SELECT I.name FROM amPortfolio a, amEmpIDept e, amLocation I WHERE a.IUserI d = e.IEmpIDeptId AND e.ILocaId = I.ILocaId

上記のクエリでは、「**ユーザ」**(SQL名:User)と「場所」(SQL名:Location) のリンクが、依然として部署と従業員または場所のテーブル内のレコードにア クセスする(リンクがない場合は主キーが「0」のレコードにアクセスする)の で、期待した結果が得られます。

#### 結果

 特に集計関数を使う場合などは、記述したクエリで前述のようなレコードを 処理することが重要になります。

#### 例

SELECT count(AssetTag) FROM amPortfolio

資産のテーブル内の資産数を数える上記のクエリを実行すると、主キーが「0」 のレコードも結果に含まれます。そのため、データベース内の実際の資産数を 求めるには、結果から1を引く必要があります。

DBMSレベルの外部結合を生成する必要はほとんどありません。

💋 注意:

実際にDBMSレベルの外部結合を処理する場合は、AQL演算子(「=\*」および 「\*=」)を使います。

### NULLの使い方

AssetCenterでは、次の2つのフィールドにのみDBMSのNULL値を使います。

- 空の「テキスト」型のフィールド
- 未入力の「日付」または「日付+時刻」型のフィールド

AQLでは、以下に示す複数のシンタックスを使うことができます。AQLはこれ らのシンタックスを、ユーザのデータベースエンジンに有効なSQLコードに変 換します。

空の「テキスト」型のフィールドについては、データベースにNULL値が保存されるので、次のいずれのシンタックスでも使うことができます。

WHERE <text field> = NULL

WHERE <text field> IS NULL

WHERE <text field> = "

未入力の「日付」または「日付+時刻」型のフィールドについては、次のシン タックスを使ってデータベースにNULL値を保存することができます。

WHERE <date or date+time field> = NULL

WHERE <date or date+time field> IS NULL

WHERE <date or date+time field> = []

#### 💋 注意:

「数値」型のフィールドが未入力の場合は、フィールド値は「0」になります。 同様に、リンクが存在しない場合は「リンク=0」または「外部キー=0」で表 します。例: Location=0 または ILocald=0

#### Self

Selfは、適用先テーブル名の表記(特殊)文字列に相当する式です。 Selfを使うとクエリが簡単になり、AssetCenterデータベースのカスタマイズを活 用できるようになります。

例

部署と従業員のテーブル名の表記文字列が次のように構成されているとします。

[Name], [FirstName], ([Phone])

AQLクエリは次のようになります。

SELECT self FROM amEmpIDept

これは、次のクエリに相当します。

SELECT (((((Name + ',') + FirstName) + '(') + Phone) + ')') FROM amEmpIDept

#### CurrentUser

CurrentUserを使うと、データベースに接続しているユーザに依存するクエリを 記述できます。

CurrentUserは、クエリの式として、またはリンクとして使うことができます。 この式はクエリエディタでは記述できないので、手動で入力する必要がありま す。

### 式として使う

例:データベースに接続している従業員が使うすべてのポートフォリオ品目を 検索します。
SELECT IPortfolioItemId FROM amPortfolio WHERE User = CurrentUser

# リンクとして使う

CurrentUserは、先ず全テーブルを検索し、次に[部署と従業員]テーブル内に ある現在のユーザのレコードに検索を絞るリンク、と見なされます。

- 「CurrentUser」形式では、この関数は現在のユーザに対応するレコードにアクセスします。
- 「CurrentUser.Field」形式では、この関数は現在のユーザのフィールド値を返します。

例:データベースに接続しているユーザがアクションをトリガした場合、別の メッセージタイプのアクションが状況に応じて発生するように設定できます。 メッセージタイプのアクションは、接続中のユーザに自動的に警告メッセージ を送信するとします。これを指定するには、アクションの詳細画面で次のよう に入力します。

🥸 説明	🖂 メッセージ	任意管理項目	履歴 書類	1
	参照オブジェクト:			-
	優先度:	普通		•
		□ 配信済通知		
受信者:	[CurrentUser.Emai	0		٩
Cc:				٩
Bec:				٩
件名:	確認メッセージ			٩
メッセージ:	アクションがトリガさ	れました。		<u>~ 9</u>
				-

# システムリストデータ

AQLクエリでシステムリストデータを使う場合は、画面に表示される値ではな く、データベースに保存されている値を使う必要があります。

例

次のクエリでは、[タイプ](SQL名:seType)フィールドが[マスターリー ス]に設定された契約を選択します。

SELECT Self FROM amContract WHERE seType = 1

[**タイプ**] (SQL名: seType) フィールドはシステムリストデータです。データベースに保存される値は次の通りです。

0:[その他]

AssetCenter

- 1:[マスターリース]
- 2:[リース明細]
- 3:[保険]
- 4: [メンテナンス]

#### 💋 注意:

AssetCenter Database Administratorを使用するか、またはデータベース構造を記述 した「Database.txt」ファイルを参照すると、システムリストデータの値を調べ ることができます。

このファイルは、「[AssetCenterのインストール先フォルダ]/doc/infos」にあ ります。

# 階層構造のテーブル

すべての階層構造のテーブルには、次のフィールドが含まれています。

- [FullName]フィールド
- [sLvl]フィールド

### [FullName]フィールド

階層構造のテーブルの各レコードには[FullName]フィールドがあります。このフィールド値の前には、親レコード(ルートまで)のフィールド値で構成されるツリー構造のパスが付きます。

パスはスペースなしのスラッシュ (/) 記号で区切られます。パスの最初と最後にもスラッシュが入ります。

例

 [資産]テーブルの[FullName]フィールドでは、次のように現在の資産の 資産タグ(CR012)の前に、親資産の資産タグが、その前に親資産の親資産 の資産タグが置かれます。

FullName = '/PC118/DD054/CR012/'

「場所]テーブルの[FullName]フィールドの場合も同様に、場所名(5階) の前に親場所名が付いた形式で保存されます。

|FullName = '/東京/府中支社/府中ビル/5階/'

# [sLvl]フィールド

階層構造のテーブルのすべてのレコードには、ツリー構造のレベルを示す[sLvl] フィールドがあります。

ルートはレベル0になります。



次のクエリは、「アジア地域営業部」のレコードとその付属部署を選択します。

SELECT Self FROM amEmpIDept WHERE (FullName LIKE '/アジア地域営業部/マ ーケティング/%') AND (sLvl >= 1)

次のクエリは、「アジア地域営業部」のレコードは選択しますが、その付属部 署は除外します。

SELECT Self FROM amEmpIDept WHERE (FullName LIKE '/アジア地域営業部/マーケティング/%') AND (sLvl = 1)

次のクエリは「アジア地域営業部」のレコードの付属部署は選択しますが、「ア ジア地域営業部」のレコード自体は除外します。

SELECT Self FROM amEmpIDept WHERE (FullName LIKE '/アジア地域営業部/マ ーケティング/%') AND (sLvl > 1)

# AQLの簡単な表記法

ここでは、AQLの記述を簡略化するための表記法を紹介します。

### 外部キー

SELECT句とORDER BY句以外の句では、最後にピリオド(.)がないリンクの SQL名は、関連付けられた外部キーのSQL名と同等に扱われます。 例えば次のクエリ

WHERE location = 0

は次のクエリと同等に処理されます。

WHERE ILocald = 0

この場合「location」は、**[部署と従業員]**テーブルを**[場所]**テーブルへつな ぐ「場所」リンクのSQL名で、「ILocald」は対応する**[資産]**テーブルの外部 キーのSQL名です。

### 表記文字列

SELECT句とORDER By句では、最後にピリオドのないSQL名は<リンクのSQL名>.self結合と同等、つまり<リンクのSQL名>.<表記文字列>と同等に扱われます。

例

部署と従業員のテーブル名の表記文字列が次のように構成されているとします。

[Name], [FirstName] ([Phone])

AQLクエリは次のようになります。

SELECT user FROM amPortfolio

これは、次のクエリに相当します。

SELECT user.self FROM amPortfolio

これ自体は次と同等です。

SELECT (((((User.Name + ',') + User.FirstName) + '(') + User.Phone) + ')') FROM amP ortfolio

## 任意管理項目

AQLでは、テーブルのフィールドに直接アクセスできるように、テーブルの任 意管理項目にも直接アクセスできます。特定のテーブルで任意管理項目の値を 検索するには、任意管理項目のSQL名の前に「fv\_」を付けます。

例:次のクエリは、**[部署と従業員]**テーブル(SQL名:amEmplDept)テーブ ルでSQL名が「fv WorkUnit」の任意管理項目の値を検索します。

SELECT fv\_WorkUnit FROM amEmpIDept

### 特殊フィールド

AQLでは、テーブルに関連付けられている特殊フィールドを活用できます。 特殊フィールドのSQL名の前に「cf\_」を付けて記述してください。

# 並べ替えとインデックス

AQLで並べ替え(ORDER BY句)を使うクエリには、以下の2通りがあります。

- クエリで指定したインデックスをAssetCenterでそのまま使うモード。この モードでは、検索結果がそのまま順次表示されます。
- クエリで指定したインデックスをAssetCenterで使わないモード。この場合は、DBMSがデータの保存方法を決定します。

#### 💋 注意:

SQL Anywhereでは、上記のいずれかを選択できるわけではありません。データベースエンジンが最適な方法を自動的に選択します。

## 例

次のクエリについて説明します。

SELECT IModelId, Brand FROM amModel ORDER BY Brand

- インデックスを使用しないアクセス:データベースエンジンは、クエリで指定した[Brand](メーカ)のインデックスを使わずに全テーブルを走査します。クエリの条件を満たすすべてのデータ項目を検索してから「メーカ」を基準にして並べ替え、ユーザに送信します。結果は一定の時間が経過しないと表示されません。
- インデックスを使用するアクセス:データベースエンジンは、[Brand]の インデックスを使って検出した結果をそのままの順番で即時表示します。このため、最初のデータ項目はすぐに表示されますが、全体を処理する時間は 長くなります。

# インデックスを使う方法

インデックスを使う方法は、クエリを作成する方法によって変わります。

# [リストの設定]メニューを使う

AssetCenterのリストごとにデータのアクセスタイプを設定できます。メインリ ストおよびタブページ内のリストで設定できます。手順は次の通りです。

- 1 設定するリストに移動します。
- 2 右クリックします。
- 3 ポップアップメニューから[リストの設定]を選択します。

4 [列/並べ替え]タブページで[インデックスの強制使用]チェックボック スをオンにし、クエリで指定されたインデックスを使って実行したクエリ結 果を、そのままの順番で即時に表示するよう指定します。別のアクセスタイ プを選択する場合は、このチェックボックスをオフにします。

## AQLを使う

AQLで直接クエリを記述する場合は、FIRST\_ROWS句でインデックスを使うことを指定できます。

例

SELECT FIRST\_ROWS AssetTag FROM amAsset ORDER BY AssetTag

#### 💋 注意:

任意管理項目テーブルの [seDataType]フィールドなどの、システムリストデー タで並べ替えを行う際にインデックスを強制すると、良好に機能しないことが あります。

# 並べ替え順

並べ替え順は次の条件に左右されます。

- データベースエンジン
- インデックス使用の有無

### Oracle

### インデックスを使う場合

- NULLのレコードは表示されません。
- ASCIIコードの値によって並べ替えるため、大文字と小文字を区別します(バ イナリソート)。

## インデックスを使わない場合

- NULLのレコードが表示されます。
- Oracleでは大文字と小文字の区別はありません。
- 例

並べ替え

AssetCenterの高度な使い方

元のリスト	A B C D a b NULL NULL
インデックスを使ったリスト	ABCDab
インデックスを使わないリスト	NULL NULL A a B b C D

### Microsoft SQL Server **t** Sybase SQL Server

並べ替え順は、データベース作成時に設定したパラメータによって決まります。 これらのエンジンでは、大文字と小文字を区別する、アクセント付きの文字も 区別するなどの設定が可能です。

### Sybase SQL Anywhere

Sybase SQL Anywhereでは、AQLクエリでインデックスを使うことはできません。

データにアクセスしてデータを並べ替える最適な方法は、データベースエンジンが決定します。

## 注意事項

複雑なクエリでは、インデックスを使う場合と使わない場合のどちらが有利で あるかを、即時に判断するのは困難です。実際に使用の有無を決定する前に、 テストを行うことをお奨めします。

特に、シンプルフィルタやクエリなどの直接的なフィルタや、アクセス制限に よる間接的なフィルタを適用したリストの場合は、インデックスのテストを行 うようにしてください。

# クエリエディタ

AssetCenterはクエリエディタを内蔵しています。このツールを使うと、クエリ の設計と結果をSQL言語でプレビューすることができます。クエリエディタは、 特にデータベース管理者やクエリに精通したパワーユーザを対象としています。

# 動作

クエリエディタでは、次のいずれかの方法でクエリを設計できます。

- グラフィカルユーザインタフェースを使って作成
- AQLで直接クエリを記述

グラフィカルユーザインタフェースを使う場合も直接AQLで記述する場合も(多くの場合、2つの方法を組み合わせて使います)、作成したクエリをSQL言語で

リアルタイムで表示することができます。ただし、クエリを直接SQLで記述す ることはできません。

図 3.1. クエリエディタの活用法



パワーユーザや管理者は、クエリエディタを使ってAQLクエリの作成、変更、 削除を実行できます。作成者や他のユーザは、作成されたクエリを状況に応じ て使います。

# クエリエディタにアクセスする

クエリエディタには次の方法でアクセスできます。

- [ツール/クエリ]メニューを使う。このメニューを使ってクエリを作成すると、他のユーザもそのクエリを自由に使用できるようになります。クエリは、次の方法で実行できます。
  - [ツール/クエリ]メニューを選択すると表示されるクエリの詳細画面 で直接実行
  - クエリのメインテーブルに表示される専用メニューから、クエリフィル タを選択して実行
- アクセス制限、クエリフィルタ、リストの設定など、AssetCenterのクエリを 呼び出す機能を使う。
- AssetCenter Exportなどの外部プログラムを使う。

次のクエリエディタでは、内容に応じてクエリを簡略化できます。 例:次のようなクエリがあるとします。

SELECT [FIRST\_ROWS] <フィールド>[, <champ>...] FROM <テーブル> [WHERE |<句>] [ORDER BY <句>]

簡易版のクエリエディタ(シンプルフィルタ、クエリフィルタなど)では、ク エリのWHERE句だけを定義するだけで済みます。クエリのその他のコンポーネ ント(クエリを実行するテーブル、フィールドなど)は暗黙的に指定します。 例えば、クエリフィルタの場合、テーブルはフィルタを適用するテーブルであ り、フィールドと並べ替え条件は**[リストの設定]**ポップアップメニューで定 義した列 / 並べ替えの条件です。**[ツール / クエリ]**メニューからクエリエディ タにアクセスした場合も同様です。

例えば、次のような明示的に記述したクエリがあるとします。

SELECT self FROM amModel WHERE Brand.Name='Compaq'

これと同じ内容のクエリを、モデルのテーブルでクエリフィルタを使って行う 場合は、次のように記述されます。

Brand.Name='Compaq'

ー方、**[リストの設定]**コマンドを使うと、次のように複雑なクエリエディタ にアクセスできます。

- [列/並べ替え]タブページでは、リストの列に表示するフィールドと並べ 替え条件を定義します(これらの並べ替え条件はORDER BY句に相当しま す)。
- [インデックスの強制使用] チェックボックスは、SQLコードのFIRST\_ROWS 句に置き換わります。
- [フィルタ(WHERE句)]タブページでは、WHERE句を定義します。
- テーブルは暗黙的に指定します。

## クエリエディタを使ってクエリを作成する

クエリエディタを使ってクエリを作成するには、**[ツール/クエリ]**メニュー を選択します。表示されるウィンドウには、次の**[フィルタ(WHERE句)]**と **[プレビュー]**タブページがあります。

- [フィルタ(WHERE句)]タブページは、クエリの条件を指定するグラフィ カルユーザインタフェースです。このタブページでSQL WHERE句の要素を 定義します。
- [プレビュー]タブページには、クエリをSQLコードに変換したものが表示 されます。このタブページでクエリをテストできます。

### 手順1: クエリの詳細画面上部のフィールドに入力する

クエリの開始テーブルを指定する必要があります。

作成するクエリに他のユーザがアクセスできるようにする場合は、[共有しない](SQL名:bPrivate)チェックボックスをオフにします。

💋 注意:

管理者は、[共有しない]オプションが選択されているクエリも含め、データ ベース内のすべてのクエリにアクセスできます。

クエリの基本的な情報を入力してから**[作成]**をクリックすると、クエリの詳細を指定するタブページを使うことができます。

## 手順2: [フィルタ(WHERE句)]タブページでフィル タ条件を定義する

AssetCenterのクエリエディタを使うと、フィールド、計算式、定数、演算子を 組み合わせる条件を定義できます。

複数のフィルタ条件を定義できます。

フィルタ条件を定義するには、次の手順に従います。

- 1 [フィールド1]に開始テーブルのフィールド、定数、または式を指定しま す(比較演算子を使う場合は[フィールド2]フィールドにも同様に指定し ます)。
- 2 ■ボタンを使ってこのクエリを画面下部のウィンドウに転送し、フィルタ条件を確認します。
- 3 [変更]をクリックし、クエリの作成を確定します。

ANDやORで連結した複数のフィルタ条件を定義するには、次の手順に従います。

- 1 前述のように、最初のフィルタ条件を作成します。
- 2 その他のフィルタ条件を定義し、[AND]ボタンまたは[OR]ボタンで条件を連結します。
- 3 [変更]をクリックし、クエリの作成を確定します。

🔰 注意:

選択した条件を変更するには、 × ボタンをクリックしてウィンドウの内容を 削除するか、またはAQLコードを直接変更します。

💋 注意:

グラフィックツールを使わずに、【フィルタ(WHERE句)】タブページの下部 のウィンドウに直接AQLでクエリを入力することもできます。

# 手順3: クエリの実行をプレビューする

クエリをテストし、SQL言語に変換したクエリを表示する手順は以下の通りです。

- 1 クエリの詳細画面の[プレビュー]タブページに移動します。
- 2 <sup>31</sup> アイコンをクリックします。クエリによって検索されたレコードが下のウィンドウに一覧表示されます。クエリ条件に一致するレコード数は、ウィンドウの右下に表示されます。

AssetCenterの高度な使い方

```
💋 注意:
```

[プレビュー]タブページに表示されるSQLコードを直接変更することはできません。

# クエリで使うフィールド

クエリのフィルタ条件を定義する場合、次のフィールドを指定できます。

- クエリの対象となるテーブル内のフィールド
- リンクしているフィールド
- テーブルに関連付けられている任意管理項目

## 式を記述する

式mを使うとクエリで計算を実行できます。例えばCount関数を使うと、クエリ によって検索されたレコード数をカウントできます。

式を記述するには、次の方法があります。

- クエリを記述するフィールドに直接入力する。
- AssetCenterの式ビルダを使う。

式ビルダを使うには、クエリの詳細画面の**[フィルタ(WHERE句)]**タブページで、編集領域の隣にある図ボタンをクリックします。



式ビルダは、次の3つの列で構成されます。

AssetCenter

- [名前]列には、既存のAQL関数のリストが表示されます。□をクリックすると、AQL関数のタイプ(集計、文字列、日付、数値、テスト)を選択できます。選択したタイプがフィルタとなり、そのタイプの関数だけを表示できます。
- [フィールド]列には、クエリに使用できるフィールドのリストが表示されます。
- [演算子]列には、式で使用できる演算子のリストが表示されます。
- [名前]、[フィールド]、[演算子]を式に挿入するには、
- 1 関数、フィールド、演算子のいずれかを選択します。
- 2 \_\_\_\_\_をクリックします。

式の定義が終了し、[OK]をクリックすると、クエリの詳細画面の[フィルタ (WHERE句)]タブページに画面が切り替り、定義した式が表示されます。

### 定数

定数 K は、選択基準に割り当てる固定の値です。例えば、メーカが「3Com」の すべてのモデルを検索する場合は、モデルのテーブルの [Brand.Name]リンク フィールドに定数値「3Com」を割り当てます。

定数を選択するには、次の手順に従います。

- 1 図アイコンをクリックします。
- 2 データベース内の既存の値を表示するウィンドウが表示され、そこから検索 条件として指定するフィールドを選択できます。

#### 💋 注意:

リストデータ型のフィールドの場合でも、<br/>
国アイコンをクリックすると上記の<br/>
選択ウィンドウが表示され、データベースで使っているリストデータの値のみ<br/>
が表示されます。

# AQL**のシンタックス**

AQLを使うには、SQL言語に関する知識が必要です。ただし、このマニュアル にはSQLのシンタックスに関する詳しい説明はありません。詳細については、 他の参考資料を参照してください。



AQLのシンタックスの表記法

AssetCenterの高度な使い方

表	3.1		表記法
---	-----	--	-----

[]	大括弧は、オプションの項目を示します。 実際に項目を入力するときは、大括弧は必 要ありません。
<>	山形括弧は、項目の短い説明を示します。 実際に項目を入力するときは、山形括弧を 使わずに、括弧内にあるテキストに該当す る情報だけを入力してください。
	縦線(パイプ文字)は、複数の項目候補を 区切るのに使います。
	省略記号は直前の内容が繰り返されること を表します。
FROM	大文字の単語はそのまま記述します。

# クエリのシンタックス

# シンプルクエリ

SELECT [DISTINCT] [FIRST\_ROWS] <選択リスト> [FROM句] [WHERE句] [GROUP BY句] [HAVING句] [ORDER BY句]

# サブクエリ

AQLでは、フィールドの代わりにサブクエリを使うことができます。

#### 💋 注意:

サブクエリでは、SELECTステートメントで式を1個だけ使うことができます。

(SELECT [DISTINCT] <式> [FROM句] [WHERE句] [GROUP BY句] [HAVING句]

🌻 警告:

)

サブクエリは括弧で囲む必要があります。

使用例

SELECT Self FROM amAsset WHERE mPrice >= (SELECT Max(mPrice)/2 FROM a mAsset)

# UNION型のクエリ

UNIONを使うと、次のように複数のクエリの結果をグループ化できます。 SELECT <選択リスト> [FROM句] [WHERE句] [GROUP BY句] [HAVING句] [UNION | UNION ALL | INTERSECTS | MINUS] SELECT <選択リスト> [FROM句] [WHERE句] [WHERE句] [GROUP BY句] [HAVING句]...] [ORDER BY句]

# クエリの要素

## フィールドとリンク

クエリには、AssetCenterデータベース内のフィールドとリンクを指定します。

フィールド名は次の方法で指定できます。

クエリの開始テーブルの場合は、そのテーブルの名前を指定する必要はありません。

[Lien. ...[Lien.]]<フィールド>

ポートフォリオ品目 (SQL名: AmPortfolio) テーブルの例

Model User.Name User.Location.Name ・参照フィールドとして指定する場合は、次のいずれかの方法でフィールドが 属するテープルの名前を指定する必要があります。 ・ FROM句でテーブルを宣言してテーブルの名前を指定します(エイリア スも可)。

<table.[link...]field>

<alias.[link...]field>

FROM句でテーブルを宣言しない代わりにコロン(:)を使います。
 <table:[link...]field>
 <table[\_alias]:[link[\_alias]...]field>

後半の2つの表記は、FROM句が使えない場合に便利です。

例えば、AssetCenterでクエリを記述する場合は、WHERE句しか使えません。ク エリの開始テーブルは暗黙的に指定されます(フィルタを適用するテーブルを クエリの詳細画面の[テーブル](SQL名:TableName)フィールドで指定する 場合は暗黙的な指定です)。これに対し、クエリで他のテーブルを使う必要が ある場合は、コロン(:)を使って明示的に指定します。

### 定数

次のシンタックスは、クエリで使用できる有効な定数です。

#### 数値定数

小数点の区切り文字としてピリオド(.)を使います。

- 例
- 12

52.23

#### テキスト型の定数

一重引用符で囲みます。

例

'Computer ' 'Monitor'

### 日付または時刻型の定数

日付または時刻型の定数は、シャープ(#)文字で囲みます。日付と時刻の形式 は、次の規則に従います。

AssetCenter

- 年は4桁で表します。
- 日付は年-月-日で表します。
- 時刻は時間-分-秒で表します。
- 24時間制を使います(A.M.またはP.M.を付けた12時間制ではありません)。
- 日付はスラッシュ (/) またはハイフン (-) で区切ります。
- 時刻はコロン(:)で区切ります。
- 月、日、時間、分、秒は、2桁で表します。
- 日付と時刻を合わせて表示する場合は、必ず日付の後に時刻を記述し、2つ をスペースで区切ります。

例

#yyyy-mm-dd hh:mm:ss#
#yyyy-mm-dd#
#hh:mm:ss#
#2004-01-01 01:00:03#

### 式

式は、次の要素で構成されます。

- 定数
- フィールド
- 関数
- サブクエリ

これらの要素に演算子や括弧を組み合わせると、複雑な式を作成できます。 比較の式のシンタックスは次の通りです。

#### <式> <比較演算子> <式>

論理式のシンタックスは次の通りです。

<比較式> <AND | OR> <比較式>

括弧を使って複数の論理式を組み合わせることができます。

### 演算子

#### 論理演算子

論理演算子を使うと、2つの式を連結できます。

#### 表 3.2. 論理演算子

演算子	説明
AND	論理積演算
OR	論理和演算

クエリを最適化するために、比較演算子を使える場合は論理演算子を使わない 方がよい場合があります。以下の例は、[割当](SQL名:seAssignment)フィー ルドが[納品待ち]または[メンテナンスのため返却]であるポートフォリオ 品目を選択するクエリフィルタを、最適化する方法です。この2つのシステムリ ストデータの値は、それぞれ「3」と「4」です。

(seAssignment=3) OR (seAssignment =4)

システムリストデータの値が「4」までの場合は、上のクエリを次のように記述 することもできます。

seAssignment >=3

#### 比較演算子

比較演算子は、2つの式を比較する時に使います。

#### 表 3.3. 比較演算子

演算子	説明
=	等しい
<>	等しくない
=!	
>	より大きい
<	より小さい
>=	
=<	
=*	右外部結合。この演算子は、AQLでリンクを処理する場合のみ使い ます。
*=	左外部結合。この演算子は、AQLでリンクを処理する場合のみ使い ます。

AssetCenter

演算子	説明
LIKE	「=」演算子と同様に機能し、ワイルドカード文字を使うこともでき
NOT LIKE	ます。
	次のワイルドカード文字を使用できます。
	パーセント(%):すべての文字列に置き換わります。
	アンダースコア(_):任意の1文字に置き換わります。
	次の指定は、データベースエンジンによって変わります ( SQL
	Anywhere、SQL Server、Sybaseはサポートしますが、Oracleはサポー トしません)。
[abc]は任意の並び文字(間にスペースがない)を定義します。	
	[a-c]は、文字列値の範囲を定義します。
	DB2では、LIKE X演算子のXにSQL列名が含まれる場合は、この演算 子を使うことはできません。この演算子で使用できるのは定数のみ です。例えば、次のようなクエリはDB2で機能しません。
	SELECT COL1, COL2 FROM TABLE1 WHERE COL1 LIKE COL2
IS NULL	フィールド値がNULLかどうかを調べます。
IS NOT NULL	AssetCenterでは、空のテキストフィールドと、値が入力されていな い <b>日付</b> または <b>日付+時刻</b> 型フィールドがNULLと見なされます。

### 💋 注意:

SQL Anywhereでは、LIKE X句のXが128文字を超える場合は処理できません。X が128文字を超えた場合にクエリを適用すると、ODBCエラーメッセージが表示 されます。このエラーは、LIKE句で[FullName]フィールドを指定して、リス トをツリー構造で表示する場合などに発生する可能性があります。

### サブクエリ専用の演算子

次の演算子を使って、サブクエリの結果と値を比較できます。

- = ANY (サブクエリ)
- = ALL (サブクエリ)
- = SOME (サブクエリ)

例

次のクエリでは、「府中支社」で使用しているメーカのポートフォリオ品目のリストを取得できます。

SELECT IModeIId, Model.Brand FROM amPortfolio WHERE Model.Brand = ANY (S ELECT Model.Brand FROM amPortfolio WHERE Location.FullName = '/府中支社')

### 選択リスト

選択リストは、抽出または表示する項目を定義します。選択リストは、クエリ 内のSELECTステートメントを指定します。 選択リストは、次のようにカンマで区切られた式で構成されます。 <式>[,<式>...]

各式をエイリアスにリンクできます。例は以下の通りです。

SELECT MrMrs, (Name + FirstName) Identity FROM amEmpIDept 選択リストは、出力クエリで出力する列名を指定する場合に特に便利です。

#### 💋 注意:

DBMSによっては、1つのSELECTステートメントに一定数の式しか記述できない場合があります。

### FROM**句**

FROM句には、SELECTステートメントの対象となるテーブルを指定します。 AQLでは、テーブル名のエイリアスを利用できます。

シンタックス

FROM <テーブル名> [テーブルのエイリアス] [, <テーブル名> [テーブルのエイ リアス>] ... ]

## クエリの開始テーブル

クエリのFROM句で最初に指定したテーブルが、クエリの開始テーブルになり ます。

テーブルが指定されていないフィールドをクエリに使った場合、AQLはその フィールドがクエリの開始テーブルに属していると見なします。AQLのFROM 句はSQLのFROM句とは異なります。

例えば次のサブクエリでは、AQLは**[資産]**(SQL名:amAsset)テーブルの **[資産タグ](**SQL**名:**AssetTag**)**フィールドを検索します。

SELECT AssetTag FROM amAsset

## クエリ内のテーブル数

クエリで指定できるテーブル数は、使用しているDBMSによって異なります。

例

- Oracle:テーブルをいくつでも使うことができます。
- Microsoft SQL ServerまたはSybase SQL Server: クエリで使えるテーブル数は 16個までです。

#### ! 警告:

特にクエリでリンクを使っている場合にクエリ内のテーブル数を数えるときは、 暗黙的に指定されているテーブルも忘れずに数えるようにしてください。DBMS のSQLで追加の結合を生成する「fv\_」表記(任意管理項目の値の検索)にも注 意してください。同様に、特殊フィールドの「cf\_」も追加の結合を生成できる 表記です。

#### 例

FROM amPortfolio

FROM amPortfolio a, amLocation I

次のクエリはすべて同じように機能します。

SELECT AssetTag FROM amAsset SELECT a.AssetTag FROM amAsset a SELECT amAsset.AssetTag FROM AmAsset

## WHERE**句**

AQLのWHERE句は、SQLのWHERE句と同様に機能します。

WHERE句は、データベースから抽出する項目を検索条件として指定します。検 索条件を記述するは、HAVING句を使うこともできます。

#### シンタックス

WHERE <検索条件>

## 検索条件の作成

多くの場合は、次の書式で条件を記述する必要があります。

<WHERE | HAVING> [NOT] <式> <比較演算子> <式><WHERE | HAVING> [NOT] <論理式>
<WHERE | HAVING> [NOT] <フィールド> [NOT] LIKE 'xxxxx'
<WHERE | HAVING> [NOT] <論理式> <AND | OR> <論理式>
<WHERE | HAVING> [NOT] <フィールド> IS [NOT] NULL

場合によっては、次のように複雑なクエリを記述する必要があります。

```
<WHERE | HAVING> [NOT] EXISTS (<サブクエリ>)
<WHERE | HAVING> [NOT] <式> [NOT] IN (<値のリスト> | <サブクエリ>)
<WHERE | HAVING> [NOT] <式> <比較演算子> <ANY | ALL> (<サブクエリ>)
```

## GROUP BY句

AQLのGROUP BY句は、SQLのGROUP BY句と同等に機能します。

#### シンタックス

GROUP BY <集合を使わない式> [, <集合を使わない式>]...

### 注意事項

GROUP BY句はテーブルのサブセットを指定する時に使います。サブセットを GROUP BY句に指定するには、フィールド名などの式を使います。

SELECTステートメントの選択リストに集合関数を使うと、GROUP BYが各サブ セットの結果の値を検索します。これで得られた結果を、HAVING句に使うこ とができます。

クエリでGROUP BY句を使うと、選択リストの各式でサブセットごとに1つの値が得られます。

#### GROUP BY - 例

次のクエリは、データベース内のメーカの総数を算出します。このクエリの場合AssetCenterは、1つのメーカに関連付けられている資産ごとに、1つのメーカインスタンスを返します。

SELECT Count(Model.Brand.Name) FROM amAsset

次のようにGROUP BY句を使うと、メーカのリストと各メーカの資産数が算出 されます。

SELECT Model.Brand.Name, count(IAstId) FROM amAsset GROUP BY Model.Brand

### HAVING**句**

AQLのHAVING句は、SQLのHAVING句と同等に機能します。

#### シンタックス

HAVING <検索条件>

## WHERE句との相違点

HAVING句は、WHERE句と同様に検索条件を指定する時に使います。ただし、 HAVING句とWHERE句は次のように異なります。

- HAVING句では、選択リスト内の集合関数に適用する制限を指定します。この場合は、クエリの結果に表示される項目数は制限されますが、集合関数にリンクしている計算には影響しません。
- クエリでWHERE句を使う場合は、検索条件によって集合関数が計算に使う 項目数は制限されますが、結果として得られる項目数に影響はありません。

#### 例

次の例に示すクエリでは、WHERE句とHAVING句が同等に機能します。 次のクエリは、名前が文字「B」で始まるメーカのリストと、メーカごとの資産 数を返します。

SELECT Model.Brand.Name, count(IAstId) FROM amAsset GROUP BY Model.Brand .Name HAVING Model.Brand.Name > 'B'

次のWHERE句を使ったクエリでも同じ結果が得られます。

SELECT Model.Brand.Name, count(IAstId) FROM amAsset WHERE Model.Brand.Na me > 'B' GROUP BY Model.Brand.Name

HAVING句を使ったクエリの例

HAVING句では集合関数(Countなど)を使うことができます。WHERE句では 使えません。集合関数を使うと、次のクエリのように、複数の資産が存在する メーカをすべて検索できます。

SELECT Model.Brand.Name, count(IAstId) FROM amAsset GROUP BY Model.Brand .Name HAVING count(Model.Brand) > 1

# ORDER BY句

AQLのORDER BY句は、SQLのORDER BY句と同等に機能します。 項目の並べ替え順には、次のタイプがあります。

- 昇順: ASC。デフォルトの並べ替え順です。
- 降順:DESC

#### シンタックス

ORDER BY <式> [ASC | DESC] [, <式> [ASC | DESC]...]

## INSERT**句**

INSERT句は、データベースのテーブルに1つまたは複数のレコードを挿入します。

#### シンタックス

INSERT INTO <テーブル名> [テーブルのエイリアス] (<フィールド名> [, <フィー ルド名>]...) VALUES ( <式> [, 式]...) | AQL**サブクエリ**)

この句はAssetCenter API AmDbExecAqlに含まれています。

AssetCenter APIの詳細については、マニュアル『プログラマーズリファレンス』の「関数の説明」の章を参照してください。

### 例

INSERT句を使うと、**受領に関する補足情報**ウィザードのコードを簡略化できます。

# INSERT**句を使用しないウィザードのコード**

hrAlarm = AmCreateRecord("amDateAlarm") IErr = AmSetFieldLongValue(hrAlarm, "bSecondLevel", 0) IErr = AmSetFieldLongValue(hrAlarm, "dtTrig1", AmGetFieldLongValue(hrAs set, 2)-IDaysBefore\*86400) IErr = AmSetFieldLongValue(hrAlarm, "IAction1Id", IActionId) IErr = AmSetFieldLongValue(hrAlarm, "IMonitObjId", IAstId) IErr = AmSetFieldStrValue(hrAlarm, "MonitoredField", "dWarrEnd") IErr = AmSetFieldStrValue(hrAlarm, "MonitoredTable", "amAsset") IErr = AmSetFieldLongValue(hrAlarm, "sDaysBefore1", IDaysBefore) IErr = AmInsertRecord(hrAlarm)

## INSERT**句を使用したウィザードのコード**

IErr = AmDbExecAqI("insert into amDateAlarm (bSecondLevel, dtTrig1, IActionId, IM onitObjId, MonitoredField, MonitoredTable, sDaysBefore1) values ( 0, " & AmGetFiel dLongValue(AmGetFieldLongValue(hrAsset, 2)-IDaysBefore\*86400 & ", " & IAstId & ", 'dWarrEnd', 'amAsset', " & IDaysBefore & ")")

## **UPDATE**句

UPDATE句は、データベースのテーブルのレコードのフィールドを更新します。

#### シンタックス

UPDATE <テーブル名> [テーブルのエイリアス] SET (<フィールド名> [, <フィー ルド名>...]) [FROM句] [WHERE句]

#### 例

UPDATE句を使うと、コマンドアクションを起動するアクションのコードを簡 略化できます。

## UPDATE 句を使用しないアクションのコード

hr = AmGetRecordFromMainId("amPOrder", [IPOrdId])
IErr = AmSetFieldLongValue(hr, "seStatus", "\$(IDS\_POSTATUS\_ORDERED)")
IErr = AmUpdateRecord(hr)

## UPDATE**句を使用したアクションのコード**

IErr = AmDbExecAqI("update amPOrder set seStatus = 21 where IPOrdId = " & [IPOrdId])

# **DUPLICATE**句

DUPLICATEは、データベースのテーブルのレコードを複製します。 この関数はAssetCenter特有です。 詳細については、マニュアル『はじめに』の「AssetCenterを初めて使用する」 の章の「レコードの処理」/「レコードを複製する」を参照してください。

#### シンタックス

DUPLICATE <テーブル名> [テーブルのエイリアス] SET (<フィールド名> [, < フィールド名>...]) [FROM句] [WHERE句]

### **DELETE**句

DELETE句は、データベースのテーブルのレコードのフィールドを削除します。

AssetCenterの高度な使い方

#### シンタックス

DELETE [FROM句] [WHERE句]

# AQLの関数

次のAQL関数は、クエリと計算式で使うことができます。

- 集合型のAQL関数
- 文字列型のAQL関数
- 日付型のAQL関数
- 数値型のAQL関数
- テスト型のAQL関数

#### 💋 注意:

使用中のDBMS専用のSQL関数を使うこともできます。この場合、他のデータ ベースエンジンにコードを移植することはできません。

# 集合型のAQL関数

#### 表 3.4. 集合型のAQL関数

機能	説明
Avg( <列> )	数値型の列項目の平均を返します。列にレコードがない
	場合は「0」を返します。
Count( <列> )	列内の非NULL値をカウントします。
Countdistinct( <列> )	列内の異なる非NULL値をカウントします。
Max( <列> )	数値、文字列、または日付型の列の最大値を返します。
	列にレコードがない場合は、0(数値型の列)、空の文字
	列(文字列型の列)、または空の日付(日付型の列)を
	返します。
Min( <列> )	数値、文字列、または日付型の列の最小値を返します。
	列にレコードがない場合は、0(数値型の列)、空の文字
	列(文字列型の列)、または空の日付(日付型の列)を
	返します。
Sum( <列> )	数値型の列値の合計を返します。列にレコードがない場
	合は、「0」を返します。

上記の関数には、GROUP BY句とHAVING句を組み合わせて使います。

# 文字列型のAQL関数

#### 表 3.5. 文字列型のAQL関数

機能	説明
Ascii( <文字列> )	<文字列>の先頭の文字のASCIIの値を返します。
Char( <n>)</n>	ASCIIコード「n」の文字を返します。
Left( <文字列>, <n> )</n>	<文字列>の先頭の「n」文字を返します。
Lower( <文字列> )	<文字列>を小文字で返します。
Ltrim( <文字列> )	<文字列>の左側のスペースを削除します。
Right( <文字列>, <n> )</n>	<文字列>の末尾の「n」文字を返します。
Rtrim( <文字列> )	<文字列>の右側のスペースを削除します。
Substring( <文字列>, <n1>,</n1>	<文字列>の第 <n1>文字から始まる<n2>文字の部分文字列</n2></n1>
<n2> )</n2>	を抽出します(<文字列>の先頭文字を第1文字としま
	す)。
Upper( <文字列> )	<文字列>を大文字で返します。

# 日付型のAQL 関数

#### 表 3.6. 日付型のAQL関数

機能	説明
Year( <日付> )	日付または日付+時刻型のフィールドの年を表す数字を返 します(1997など)。
Month( <日付> )	日付または日付+時刻型のフィールドの月を表す数字を返 します(1-12)。
Day( <日付> )	日付または日付+時刻型のフィールドで、その月の何日目 かを表す数字を返します(1-31)。
DayOfYear( <日付> )	日付または日付+時刻型のフィールドで、その年の何日目 かを表す数字を返します(1-366)。

機能	説明
WeekDay( <日付> )	日付または日付+時刻型のフィールドの曜日を表す数字を 返します。
	この数字は、サーバの設定によって変わります。例えば、
	SybaseまたはMicrosoft SQL Serverのデフォルトの設定は
	(1=日, 2=月,, 7=土)、Oracleのデフォルトの設定は(1= 月,, 7=日)です。
Hour( <時間> )	時刻または日付+時刻型のフィールドの1日の時間を表す
	数字を返します(0-23)。
Minute( <時間> )	時刻または日付+時刻型のフィールドの分を表す数字を返 します (0-59)。
Second( <時間> )	時刻または日付+時刻型のフィールドの秒を表す数字を返
	します (0-59)。
Getdate()	サーバの現在のシステム日付を返します。
AddDays( <日付>, <数値> )	日付または日付+時刻型のフィールドに特定の日数を加え
	<u>ます。</u>
AddHours( <日付>, <数値> )	日付または日付+時刻型のフィールドに特定の時間数を加 えます。
AddMinutes(<日付>,<数值> )	日付または日付+時刻型のフィールドに特定の分数を加え ます。
	日付または日付+時刻型のフィールドに特定の秒数を加え ます。
	日付1から日付2までの日数(10進の浮動小数点数)
HoursDiff(<日付1>,<日付2> )	日付1から日付2までの時間数(10進の浮動小数点数)
/ MinutesDiff( <日付1>, <日付 2> )	日付1から日付2までの分数(10進の浮動小数点数)
SecondsDiff(<日付1>,<日付 2>)	日付1から日付2までの秒数(10進の浮動小数点数)
DbToLocalDate( <日付> )	データベースサーバのタイムゾーンの日付を、クライア
· · ·	ントマシンで定義したタイムゾーンの日付に変換します。
LocalToDbDate( <日付> )	クライアントマシンのタイムゾーンの日付を、データベー
	スサーバのタイムゾーンの日付に変換します。

### 表 3.7. 日付型のAQL関数

説明	AssetCenter <b>クエリ言語</b>
先週変更したすべてのレコード	AddDays( dtLastModif,7 )>=Getdate()

説明	AssetCenter <b>クエリ言語</b>
過去1時間に通知されたすべての作業指示	HoursDiff( Getdate(), dtNotif ) <= 1
	または
	AddHours( dtNotif, 1 ) >= Getdate()
過去30分間に通知されたすべての作業指示	MinutesDiff( Getdate(), dtActualFixed ) <= 30
	または
	AddMinutes( dtActualFixed, 30 ) >= Getdate()
次のクエリは、オープンした日付る 検索します。クライアントマシンの	ヒクローズした日付が同じである作業指示を Dタイムゾーンを使います。
SELECT Self FROM amWorkorder W Start)) = DayOfYear(DbToLocalDate(	/HERE DayOfYear(DbToLocalDate(dtActualFix dtActualFixed))
	プンした作業指示を検索します。

SELECT Self FROM amWorkorder WHERE DayOfYear(DbToLocalDate(dtActualFix Start)) = DayOfYear(DbToLocalDate(GetDate()))

# 数値型のAQL関数

#### 表 3.8. 数値型のAQL関数

機能	説明
Abs( <数値> )	「数値」の絶対値を返します。
Ceil( <数値> )	「数値」以上で一番小さい整数を返します。
Floor( <数値> )	「数値」以下で一番大きい整数を返します。
Mod( <a>, <b> )</b></a>	「a」を「b」で割った余りを返します(a=qb+r、ただし
	qは整数で0 <= r < q)。
Round( <a>, <n> )</n></a>	「a」を小数点以下第「n」位で四捨五入します。
Trunc( <a>, <n> )</n></a>	「a」を小数点以下第「n」位で切り捨てます。

#### 適用例

Abs (2.516) = 2. Ceil (2.516) = 3. Floor (2.516) = 2. Mod (6,4) = 2. Round (31.16, 1) = 31.20. Round (31.16, -1) = 30.00.

AssetCenterの高度な使い方

Round (31.16, -1) = 30.00. Trunc (31.16, 1) = 31.1.

# テスト型のAQL関数

#### 表 3.9. テスト型のAQL関数

機能	説明
IsNull( <a>, <b> )</b></a>	「a」がNullの場合は「a」を「b」で置き換えます。「a」
	と「b」のデータ型に互換性がなければなりません。

# クエリの例

本節では、各例につきクエリ作成の1側面が説明されています。以下の例を参考 にして独自のクエリを作成してください。

ここで紹介する例では、完全なシンタックスでクエリを記述しています。これ らのクエリを実際にテストする場合は、AssetCenter Exportを使います。これら のクエリをデータベースのクエリフィルタなどで使用するには、シンタックス を変更する必要があります。

例えば、次のような明示的に記述したクエリがあるとします。

SELECT self FROM amAsset WHERE Model.Brand.Name='Compaq'

これを資産のテーブルのクエリフィルタで使うには、次のようにWHERE句だけ を明示的に指定したシンタックスに変更します。

Model.Brand.Name='Compaq'

AssetCenterプログラムに付属しているデモ用データベースに格納されているクエリも、参照してください。

#### 💋 注意:

使用中のDBMSのSQLコードに書き換えられたクエリは、クエリの詳細画面の [プレビュー]タブページに表示されます。

# メインテーブルのフィールドを特定の値と比較する

例:メーカが「Compaq」であるすべてのポートフォリオ品目

SELECT Self FROM amPortfolio WHERE Model.Brand.Name = 'Compaq'

# メインテーブルのリンクを別のリンクと比較する

例:親資産と場所が同じであるすべてのポートフォリオ品目

SELECT Self FROM amPortfolio WHERE Location = Parent.Location

# メインテーブルのリンクを特定の値と比較する

例:「大阪支社」に直接リンクしているすべての部署と従業員

SELECT Self FROM amEmpIDept WHERE Parent.Name = '大阪支社'

# メインテーブルにリンクしたテーブルのフィールドの 値に応じて比較する

例:親資産と場所名が同じであるすべてのポートフォリオ品目

SELECT Self FROM amPortfolio WHERE Location.Name = Parent.Location.Name

## 階層構造のテーブル

## [FullName]フィールドを使う

例:名前が「大阪支社」である場所の、すべての関連場所

SELECT Self FROM amLocation WHERE FullName LIKE '/大阪支社/%'

# [FullName]フィールドと[sLvl]フィールドを使う

階層構造のテーブルのクエリでは、[FullName]フィールドと[sLvl]フィールドが頻繁に使用されます。

例:「大阪支社」場所に関連する従属場所で、階層のレベルが3より低いレコー ド

AssetCenterの高度な使い方

ツリー構造のルートの階層レベルは「0」です。

SELECT Self FROM amLocation WHERE (FullName LIKE '/大阪支社/%') AND (sLv | < 3)

フルネームの先頭と末尾にスラッシュ(/)記号が付くことに注意してください。

# 2つの条件を組み合わせたクエリ

例:役職が「会計部長」で場所「府中支社」に位置するすべての従業員

SELECT Self FROM amEmpIDEpt WHERE (Title = '会計部長') AND (Location.Nam e = '府中支社')

# 数値、日付、テキスト型フィールドの比較

例:2003年1月1日から2003年12月31日までに実行したすべての作業指示

SELECT self FROM amWorkOrder WHERE (dtActualFixStart >= #2003-01-01 00:00: 00#) AND (dtActualFixStart <= #2003-12-31 00:00:00#)

## 任意管理項目に関するクエリ

例:[fv\_Size](SQL名)任意管理項目の値が150 cm以上であるすべてのポー トフォリオ品目

SELECT Self FROM amPortfolio WHERE fv\_Size >= 150.00

## 式でレコードを検索する

例:購入価格が、データベース内で最高の購入価格に等しいすべての資産。メ インクエリ内にサブクエリを使って最高価格を指定します。

SELECT Self FROM amAsset WHERE mPrice = (SELECT max(mPrice) FROM amA sset)

# 未入力のフィールドを検索する

例:電話番号のないすべての従業員。空の文字列を一重引用符で囲みます。

SELECT Self FROM amEmpIDept WHERE Phone="

# リンクなしのレコードを検索する

### 1リンクの場合

例:ユーザに割り当てられていないすべてのポートフォリオ品目。リンクがないことを「0」で表します。

SELECT Self FROM amPortfolio WHERE User = 0

### n**リンクの場合**

例:資産にリンクしていないすべてのモデル

SELECT self FROM amModel WHERE NOT ( EXISTS (SELECT A1.IAstId FROM a mAsset A1 WHERE A1.IModeIId = amModel.IModeIId))

上記のクエリは、モデルのテーブルを走査して各モデルに属する資産の数を0と 比較します。

# 1リンクとnリンクのテストを組み合わせた例

例:親モデルも従属モデルもないすべてのモデル

SELECT self FROM amModel WHERE (NOT ( EXISTS (SELECT A1.IModeIId FRO M amModel A1 WHERE A1.IParentId = IModeIId))) AND (Parent = 0)

上記のクエリでは次の処理を行います。

- 1リンク(Parent = 0)のテスト。親資産のないモデルを選択します。
- nリンク(0=(SELECT COUNT(a.IModelId) FROM amModel a WHERE a.IParentId = IModelId))のテスト。従属モデルのないモデルを選択します。nリンクの テストは、各モデルの識別子[IModelId]を選択し、[IParentId]識別子が [IModelId]に等しいすべてのモデルをカウントします。

# 別の組み合わせの例

「コンピュータ」属性の従属モデルがないすべてのモデル

SELECT self FROM amModel p WHERE NOT ( EXISTS (SELECT IModeIId FROM amModel WHERE (FullName LIKE (p.FullName + '%/')) AND (Nature.Name = 'コン ピュータ')))

💋 注意:

AssetCenter Exportを使ってこのクエリを試行すると、エラーメッセージが表示 されますが、これは無視してください。クエリは正しく機能しています。

# エイリアスを使ったクエリ

例:「Peregrine」研修プログラムと「データベース」研修プログラムを受講したすべての従業員

開始テーブル:[部署と従業員]テーブル

クエリは次の通りです。

SELECT Self FROM amEmpIDept WHERE (Trainings\_1.Title = 'Peregrine') AND (Tr ainings\_2.Title = 'データベース')

Training\_1とTraining\_2と表記したエイリアスによって、Trainingリンクで関連付けられている2つの異なるレコードに2つの条件を定義できます。

以下のように記述すると、

SELECT Self FROM amEmpIDept WHERE (Trainings.Title = 'Peregrine') AND (Train ings.Title = 'データベース')

クエリは、2つの名前がついた1つの研修コースを受講したすべての従業員を選択します。

以下のように記述すると、

SELECT Self FROM amEmpIDept WHERE (Trainings.Title = 'Peregrine') OR (Trainin gs.Title = 'データベース')

クエリは、2つの名前の内いずれかの研修コースを1つ受講したすべての従業員 を選択します。



本章では、AssetCenterの書式を作成する方法について説明します。 書式のリストを表示するには、[ツール/レポート機能/書式]メニューを選 択します。



書式は、データを印刷するための書類様式です。 書式は、Seagate Crystal Reportsとは異なり、AssetCenter内で直接作成できます。

# 既製の書式を実際に使用するデータベースにインス トールする

AssetCenterには既製の書式が付属しています。書式はデモ用データベースにインストールされていますが、実際に使用するデータベースを作成または更新する際に、書式をこの本番用データベースに挿入する必要があります。

# データベース作成時の書式のインポート

データベース作成時に書式をインポートするには、

- 1 AssetCenter Database Administratorを起動します。
- 2 [ファイル/開く]メニューを選択します。
- 3 [データベース記述ファイルを開く(新規データベースの作成)]オプショ ンを選択します。
- 4 AssetCenterのインストール先フォルダのサブフォルダ「config」に格納され ている「gbbase.dbb」を選択します。
- 5 [アクション/データベースの作成]メニューを選択します。
- 6 **『管理』**マニュアルの、「AssetCenter**データベースの作成」**の章の指示に 従ってフィールドに値を入力し、データベースを作成します。
- 7 [インポートシナリオの使用]オプションを選択します。
- 8 [インポートするデータ]枠内で書式を選択します。
- 9 [作成]をクリックし、データベースの作成を確定します。

# 既存のデータベース内に書式をインポートする

書式を既存のデータベースにインポートするには、

- 1 AssetCenter Database Administratorを起動します。
- 2 [ファイル/開く]メニューを選択します。
- 3 [データベース記述ファイルを開く(新規データベースの作成)]オプショ ンを選択します。
- 4 AssetCenterのインストール先フォルダのサブフォルダ「config」に格納され ている「gbbase.dbb」を選択します。
- 5 [**アクション / データベースの作成**]メニューを選択します。
- 6 [**データベース**]フィールドで、データベースを選択します。
- 7 [データベースの作成]と[システムデータの作成]オプションをオフにします。
- 8 [インポートシナリオの使用]オプションを選択します。
- 9 [インポートするデータ]枠内で書式を選択します。
- 10 [作成]をクリックし、書式のインポートを確定します。

# 書式を作成する

書式を作成するには、[ツール/レポート機能/書式]メニューを選択して、 書式のリストを表示します。
基本的な情報

- 1 書式の名前を入力します。
- 2 書式のタイプ(リストまたは詳細)を選択します。 どちらのタイプの書式にも、テキストと事前に定義した画像を入力できます。

2つのタイプは、次の点が異なります。

- リスト:フィルタを適用するなどして現在表示しているレコードのリストを印刷します。印刷されるデータは、現在表示している列項目のデータです。
- 詳細:選択している1個のレコードの詳細画面のフィールド(例:資産の 詳細画面)の値と、そのレコードにリンクしているレコードのリスト (例:その資産の付属品)を印刷できます。
- 3 書式のメインテーブルを選択します。

#### ! 警告:

[テーブル] (SQL名: TableName) フィールドでテーブルを指定すると、作成 した書式がそのテーブルのリストだけに適用されます。

# 書式とオブジェクトを編集する

書式を編集するには、まずオブジェクトを定義してから、そのオブジェクトを ページ内に配置します。

- ページに新しいオブジェクトを挿入するには、次の手順に従います。
- 1 [書式]タブページを選択します。
- 2 [**書式**]タブページの左側に表示されるオブジェクトのアイコンをクリック します。

アイコン	機能
•	書式内のオプジェクトを選択します(変更する場合など)。
Α	印刷するレコードに依存しない固定テキストと変数(現在の日付など) を追加します。
	画像を追加します。
f(x)	フィールド値と固定テキストの文字列を含む計算式を挿入します。

アイコン	機能
k	書式内のオブジェクトを選択します(変更する場合など)。
	レコードのリストを挿入します。
	このツールを使うと、ページ内にリストを配置できます。
	詳細タイプの書式では、印刷するレコードとフィールドを含んでいるリ
	ンクテーブルを定義できます。

- 3 ページ内にマウスポインタを置きます。
- 4 ページ内でクリックします。
- 5 マウスでドラッグしてフレームを描きます。このフレームはオブジェクトを 配置するためのスペースです。
- オブジェクトを配置するフレームをダブルクリックします。オブジェクトの プロパティを指定できるウィンドウが表示されます。
- 7 オブジェクトのプロパティを定義します。
- 8 [変更]をクリックします。
- 書式には次のオブジェクトを挿入できます。

### 固定テキスト

固定テキストは、印刷するレコードに依存しないテキストです。様々な文字や 次に示す変数と組み合わせることができます。

- \$D:印刷日
- \$U:書式を印刷するAssetCenterユーザの名前
- \$C:ページ番号

\$N:印刷するページ総数

🌻 警告:

テキストは引用符で囲まないでください。

例

印刷日:\$D 担当者:\$U

#### 計算式

計算式は、詳細タイプの書式でのみ使用できます。

- 計算式で次の情報を組み合わせることができます。
- AssetCenter データベースのフィールド値

AssetCenterの高度な使い方

• 引用符で囲んだ固定テキスト

例

"資産:" AssetTag " / " Brand

計算式を使って計算することはできません。

# リスト

- リストタイプの書式:1つのリストを挿入できます。このリストは、[ファ イル/印刷]メニューを選択した時にアクティブウィンドウのリストと置き 換わります。
- 詳細タイプの書式:リストの数に制限はありません。リストには、現在のレ コードに関連するすべてのレコード(資産のすべての付属品など)が表示さ れます。

#### 画像

画像(ロゴなど)を挿入できます。

#### 💋 注意:

リストタイプの書式を作成する場合は、印刷するフィールドを選択できません。 指定したテーブルのリストに表示される列項目のフィールド値が印刷されます。 詳細タイプの書式を作成する場合は、印刷するフィールドを1つずつ選択できま す。

# 書式内のオブジェクトのプロパティ

### 位置とサイズ

オブジェクトの位置を変更するには、オブジェクトをドラッグします。オブジェ クトのサイズを変更するには、オブジェクトの端をドラッグします。

#### 💋 注意:

複数のオブジェクトを同時に選択すると、オブジェクトの位置とサイズを一度 に変更できます。複数のオプジェクトを一度に選択するには、オブジェクト全 体の周りをマウスでなぞって選択フレームを描きます。[Ctrl]キーを押したま ま、オブジェクトを1つずつ連続してクリックして選択する方法もあります。

### プロパティ

オブジェクトをダブルクリックするか、または**[内部用書式/プロパティ]**メ ニューを選択すると、選択したオブジェクトのプロパティのリストが表示され ます。

プロパティリストには2つの列があります。1つ目の列にはにプロパティ名が表示され、2つ目の列ではプロパティの値を変更できます。

プロパティを変更するには、2つ目の列をクリックします。

単純なプロパティ(テキスト、計算式、リスト、背景色、文字の色、行揃え、 オブジェクトの配置)の場合は直接変更できます。複雑なプロパティ(テキス トや計算式のフォント、枠、リストの内容、画像)の場合は、別のウィンドウ が表示されます。

#### テキスト

テキストを直接入力します。

#### 計算式

計算式の特殊文字列ウィンドウには、計算式を簡単に作成できるように、選択 している書式のタイプと互換性のあるフィールドだけをツリー構造で表示する ドロップダウンリストがあります。ツリー内のノードをクリックすると、現在 のフィールドがリスト内で選択したフィールドと置き換わります。複数のフィー ルドを連続して指定するには、各フィールドを2重引用符で囲みます。

### 背景色、文字の色

16色から選択できます。

#### 行揃え

フレーム内のテキストの行揃えを指定します。ドロップダウンリストから、左 揃え、中央揃え、または右揃えを選択できます。

### ページ内の配置

ページ内のテキストの横の位置を指定します。ドロップダウンリストから次の 位置を選択できます。

- 左揃え
- 右揃え
- 中央揃え
- 両端揃え:この場合は、書式作成画面で定義した位置を保持します。

#### フォント

オブジェクトのフォントとサイズを選択するには、プロパティ画面の編集セル の右端にある 図ボタンをクリックします。

#### 枠

オブジェクトの周囲に境界線を追加するには、プロパティ画面の編集セルの右 端にある Mボタンをクリックします。罫線の選択画面が表示されます。 3D効果を選択すると、境界線に浮き出し効果を付けることができます。 3D効果を使わない場合は、上下左右の境界線、境界線の色および太さを選択で きます。

#### Image

画像を挿入するには、次の手順に従います。

- 1 プロパティ画面の編集セル内のボタンをクリックします。
- 2 [画像の選択]画面でグラフィックファイルを選択します。

#### リスト内のリンク

書式内に表示するリストを選択するには、ドロップダウンリストから希望する リスト(従業員が使う資産のリストなど)を選択します。

#### リストの内容

💋 注意:

リストの内容は、詳細タイプの書式を編集する場合にのみ設定できます。

リストの内容を定義するには、プロパティ画面の編集セル内のボタンをクリックします。設定ウィンドウが開き、リスト内のすべての列項目が表示されます。 各列項目ごとに、次の内容を定義できます。

AssetCenter

- 列のタイトル
- 列の内容を定義する計算式
- 列のサイズ(その列がリスト内に占めるパーセント)
- 列のタイトルと内容の行揃え
- 列のタイトルまたは内容の個々の行揃え
- 縦と横の境界線

リストから列を削除するには、[Delete]キーを使います。

リストに列を挿入するには、リストの最後の列項目の次のセルに新しい値を入 力します。

各セルは、プロパティのリストと同じ原理で編集できます。

### グリッドを使う

グリッドは、画面の背景に表示される縦と横の目盛線です。オブジェクトの配置に使います。

書式の詳細画面の**[内部用書式 / グリッド]** ポップアップ メニューを使って、 次の操作を行うことができます。

- グリッドの表示と非表示の切り替え
- グリッドの目盛幅の定義

横と縦の線の交点のみが表示されます。線の間隔によって、ページにオブジェクトを配置するときの精度が決まります。

### 書式のページ設定

書式の詳細画面の**[書式/ページ設定]**ポップアップメニューを使って、次のフォーマットを定義できます。

- ページサイズ
- 縦または横のレイアウト
- 書類のマージン(余白)
- ヘッダーとフッター

#### 💋 注意:

書類の余白部分にあたるフッターおよびヘッダーのサイズは、編集ゾーンで直 接変更できます。マージンマーカーや点線で表示されるヘッダーまたはフッター ゾーンの境界線をドラッグしてください。

ページのヘッダーとフッターにテキストを挿入するには、次の手順に従います。

- 1 [書式/ページ設定]メニューを選択します。
- 2 [**ヘッダー**]および[**フッター**]チェックボックスをオンにします。
- 3 [OK]をクリックします。
- 4 書式のヘッダーゾーンまたはフッターゾーンに移動します(ヘッダーゾーン とフッターゾーンは点線の境界線で示されます)。
- 5 オブジェクトをこれらのゾーンで直接挿入するか、または本文のページゾー ンからオブジェクトを移動します。
- 6 [変更]をクリックして変更を確定します。

#### 図 4.1. 書式のヘッダー



💋 注意:

ヘッダーゾーンとフッターゾーンから本文のページゾーンにオブジェクトを移 動することはできません。

# 定期レポートを簡単に作成する方法

定期的に必要なレポートを作成するには、次の方法をお奨めします。

- 1 必要な情報を表示できるビューを作成します。
- 2 このビューを書式にリンクします。

ビューには、次の情報を記憶させることができます。

- 並べ替えの基準
- 適用するフィルタとフィルタの値
- 表示する列項目のリスト

書式を使って、レポートのページレイアウトを設定します。

レポートを印刷するには、

- 1 事前に作成したビューを表示します([**ツール/ビュー**]メニューを使いま す)。
- 2 表示したビューから印刷を行います([ファイル/印刷]メニューを使います)。このときに、必ずレポートの適切なタイプと適切な書式を選択します。

### モジュールに関連する書式の識別

あるモジュールに関連する書式を識別するには、

- 1 AssetCenterを起動します。
- 2 書式のリストを表示します([ツール/レポート機能/書式])。
- 3 リスト内で右クリックします。
- 4 [リストの設定]を選択します。
- 5 リストの列見出しにリンク[**ドメイン**] (Domain)を追加します。
- 6 [OK]をクリックします。
- 7 [ドメイン]列でリストを並べ替えます。
- 8 モジュールの書式はドメインの名前で識別できます。 例:/**調達/書式**/



本章では、AssetCenterでレポートを編集する方法について説明します。 レポートのリストを表示するには、[ツール/レポート機能/レポート]メ ニューを選択します。

# レポート作成プログラムの操作とインストール



AssetCenterでは、レポート作成プログラムであるCrystal Reportsを使います。このソフトウェアで作成したレポートのファイル拡張子は「.rpt」です。

#### 💋 注意:

使用可能なレポートのリストとコメントは、AssetCenterのインストール先フォ ルダの「datakit *l*standard *l*reports」サブフォルダの「reports.txt」ファイル内にあり ます。

Crystalレポートは、 /datakit /standard /reports /rptフォルダにあります。

「reports.txt」ファイルはレポートのインポートスクリプトに使用されるため、 このファイルの構造は変更しないでください。ただし本番用データベースへ一 部のレポートのみをインポートする場合は、インポートスクリプトを実行する 前に、このファイルの**全部**の行を削除できます。ここに別のレポートを追加す ることもできます。

### 既存のレポートの印刷にフルバージョンのCrystal Reports は不要

AssetCenterのインストール時に適切なオプションを選択すれば、限定版のCrystal Reportsが一緒にインストールされます。

AssetCenterデータベースの現在のデータを既存のレポートを使ってプレビューおよび印刷するには、この限定版のCrystal Reportsで十分です。

### 既存のレポートの変更や新しいレポートの作成にはフル バーションのCrystal Reportsが必要

AssetCenterで直接新しいレポートを作成することはできません。

新規レポートを作成するには、Crystal Reportsの完全バージョンをインストール する必要があります。制約点は以下の表の通りです。

	日本語版:バージョン7.0
サポートされるCrystal Reportsの言語	• フランス語
	• 英語
	• ドイツ語
	• スペイン語
	• イタリア語
	<ul> <li>日本語</li> </ul>
	• ポーランド語
	Crystal ReportsとAssetCenterではそれぞれ別 の言語を使用できます。

サポートされるCrystal Reports バージョン バージョン7.xと8.0

Crystal Reports **にサポートされる**ODBC **ドラ** ODBC AssetCenter ドライバ イバ

# Crystal Reports 完全版またはランタイムのインストール

AssetCenterのインストールプログラムでは、Crystal Reportsランタイムバージョ ンをインストールできます。既存のレポートを印刷するにはこのバージョンで 十分です。AssetCenterのインストール時に適切なオプションをオンにします。 既にランタイムバージョンがインストールされている場合、またはCrystal Reports の完全版をインストールする場合は、ランタイムバージョンをインストールす る必要はありません。

### 既製のCrystal Reports レポートを実際に使用するデー タベースにインストールする

AssetCenterには既製のレポートが付属しています。レポートはデモ用データベー スにインストールされていますが、実際に使用するデータベースに挿入する必 要があります。

#### 💋 注意:

既製レポートはAssetCenterの以下のバージョンには付属していません。

- 日本語版
- ポーランド語版

レコードを1つずつ挿入するには、

- 1 AssetCenterを起動します。
- 2 実際に使用するデータベースを開きます。
- 3 [ツール/レポート機能/レポート]メニューを選択します。
- 4 新しいレポートを作成します。
- 5 レポートの詳細画面で[ファイル]タブページを選択します。
- 6 [**インポート**] ボタンをクリックします。
- 7 AssetCenterのインストール先フォルダのサブフォルダ datakit *s*tandard *k*eports *k*pt で、必要な「.rpt」拡張子のファイルを選択します。

### 新規データベース作成時のレポートのインポート

#### データベース作成時にレポートをインポートするには、

- 1 AssetCenter Database Administratorを起動します。
- 2 [ファイル/開く]メニューを選択します。
- 3 [データベース記述ファイルを開く(新規データベースの作成)]オプショ ンを選択します。
- 4 AssetCenterのインストール先フォルダのサブフォルダ「config」に格納され ている「gbbase.dbb」を選択します。
- 5 [アクション/データベースの作成]メニューを選択します。
- 6 **『管理』**マニュアルの、「AssetCenter**データベースの作成」**の章の指示に 従ってフィールドに値を入力し、データベースを作成します。
- 7 [インポートシナリオの使用]オプションを選択します。
- 8 [インポートするデータ]枠内でCrystal Reports レポートを選択します。
- 9 [作成]をクリックし、データベースの作成を確定します。

### 既存のデータベース内にレポートをインポートする

- レポートを既存のデータベースにインポートするには、
- 1 AssetCenter Database Administratorを起動します。
- 2 [ファイル/開く]メニューを選択します。
- 3 [データベース記述ファイルを開く(新規データベースの作成)]オプショ ンを選択します。
- 4 AssetCenterのインストール先フォルダのサブフォルダ「config」に格納され ている「gbbase.dbb」を選択します。
- 5 [**アクション / データベースの作成**]メニューを選択します。
- 6 [**データベース**]フィールドで、データベースを選択します。
- 7 [データベースの作成]と[システムデータの作成]オプションをオフにします。
- 8 [インポートシナリオの使用]オプションを選択します。
- 9 [インポートするデータ]枠内でCrystal Reports レポートを選択します。
- 10 [作成]をクリックし、レポートのインポートを確定します。

### レポートの詳細設定

レポートのリストを表示するには、**[ツール/レポート機能/レポート]**メ ニューを選択します。 AssetCenterのレポートの詳細画面で、次の情報を指定します。

### ファイル

このフィールドを直接編集することはできません。このフィールドには、**[インポート]**を使ってインポートしたレポートのファイルの完全名(ファイルへのパスと拡張子付き)が入ります。

次に、レポート機能に関連するボタンについて説明します。

- [インポート]: このボタンは、レポートの詳細画面に表示されます。この ボタンをクリックすると外部のレポートをインポートできます(初回はレ ポートの作成、2回目以降はレポートの変更として処理されます)。外部の レポートのファイル拡張子は「.rpt」です。外部のレポートをインポートす ると、レポートの詳細画面の[ファイル](SQL名: FileName)フィールド にそのファイルのファイル名が入ります。
- [出力]:このボタンは、レポートの詳細画面に表示されます。このボタン をクリックすると、AssetCenterデータベース内のレポートから「.rpt」形式の ファイルを作成できます。デフォルトでは、ダイアログボックスが開き、 [ファイル]フィールドに入っているファイル名が表示されます。「.rpt」 形式のファイルを作成すると、外部のレポート作成プログラムを使ってレ ポートを変更できます。
- [プレビュー]:このボタンにアクセスするには、[ファイル/印刷]メニューを選択します。このボタンをクリックすると、レポートを印刷する前にプレビューすることができます。
- [印刷]:このボタンにアクセスするには、[ファイル/印刷]メニューを 選択します。このボタンをクリックするとレポートを印刷できます。

#### 💋 注意:

[プレビュー]ボタンまたは[印刷]ボタンをクリックすると、AssetCenterは データベース内のレポートから一時ファイルを作成します。このファイルは Crystal Reportsの印刷エンジンで処理されます。処理されると、一時ファイルは 直ちに消去されます。表示または印刷されるデータは、現在データベース内に 存在するデータです。

### Seagate Crystal Reportsのレポートを変更する

AssetCenterデータベース内のレポートを変更するには、Crystal Reportsプログラムが必要です。

レポートを変更するには、次の手順に従います。

- AssetCenterで[ツール/レポート機能/レポート]メニューを選択して、レポートの詳細画面を表示します。
- 2 [出力]ボタンをクリックし、「.rpt」形式のファイルとして出力します。

- 3 出力した「.rpt」レポートをCrystal Reportsで変更して保存します。
- 4 AssetCenterで[ツール/レポート機能/レポート]メニューを選択して、も う一度レポートの詳細画面を表示します。
- 5 変更した「.rpt」ファイルをインポートして更新し、レコードを変更します。

### Crystal Reports 統計

自動的に更新されるCrystal Reportsレポートを表示するには、[ツール/レポー ト機能/Crystal Reports統計]メニューを選択します。

[ツール/レポート機能/レポート]メニューを使って表示するのと同じレポートを表示できます。

#### 属性

表示するレポートの属性を指定します。このフィールドの右のフィールドで、 特定のレポートを選択できます。選択できるレポートは、選択する属性によっ て異なります。

# [自動更新]ボタン

このボタンのアイコンは、 🕫 です

- このボタンをクリックすると、レポートが更新されます。
- このボタンをマウスの右ボタンでクリックすると、レポートを自動的に更新 する頻度を定義できます。

# [ズーム] ボタン

このボタンのアイコンは、<sup>40</sup>です。 拡大 / 縮小率 (3段階)を変更します。

### 詳細レポートを作成する

詳細レポートは、リストで選択したレコードの詳細情報を印刷するレポートです。

### 利用例

- 1 資産のリストを表示します。
- 2 資産を選択します。
- 3 [ファイル/印刷]メニューを選択します。
- 4 [タイプ]フィールドを[詳細レポート(Crystal Reports)]に設定します。
- 5 レポートを選択します。
- 6 印刷します。
- これで、選択したレコードごとの詳細レポートが作成されます。

# Crystal Reportsでレポートを設定する

詳細レポートを作成するには、次の手順に従います(Crystal Reports Professional 5.0を使用した場合)。

1 [Insert/Formula Field]メニューを使って、計算式フィールドを作成します。 次のシンタックスに従ってフィールド名を入力します。

<レポートに使うテーブルのSQL名>Id

💋 注意:

テーブルのSQL名の大文字と小文字を区別して入力してください。

例えば、[資産]テーブルのレポートを作成する場合は、次の計算式を使い ます。

amAssetId

#### 💋 注意:

計算式のフィールド名のシンタックスと主キーのSQL名を混同しないでくだ さい。例えば、[資産]テーブルの主キーは[IAstId]で、[amAssetId]で はありません。

計算式[CurrentUserId](大文字と小文字を区別)を使って、レポートを印 刷するユーザを特定します。印刷時に、この計算式がAssetCenterデータベー スに接続しているユーザのID番号(現在のログインのSQL名: IEmpIDeptIdの 値)を取得します。

指定したテーブルの特定のレコードのレポート結果を表示する場合は、計算 式フィールドを編集し、AssetCenterデータベース内にあるそのレコードの テーブルの主キーを入力します。

例

#### 512

#### 💋 注意:

新しい計算式フィールドの名前を確定すると表示されるウィンドウでフィー ルドを編集します。計算式フィールドが既に存在する場合は、[Edit]ボタ ンをクリックして編集します。

[Report/Edit Selection Formula/Record]メニューを使って、選択計算式を編集します。この式には、次のシンタックスを使います。

{<対象テーブルのSQL名>.<主キーとして使うフィールドのSQL名>}=@<計 算式フィールド名>}

ここではテーブルとフィールドのSQL名の大文字と小文字の区別は重要では ありません。

{amAsset.IAstId} = {@amAssetId}

このようにして設定したレポートをAssetCenterのデータベースにインポートすると、自動的に対象レポートとして指定されます。これを確認するには、次の 手順に従います。

- 1 **[ツール/レポート機能/レポート]**メニューを使ってレポートのリストを 表示します。
- 2 新しいレポートを作成します。
- 3 [インポート]ボタンをクリックして、前述の手順で設定したCrystal Reports ファイル(拡張子は「.rpt」)をインポートします。
- 4 このファイルが追加されると、[テーブル] (SQL名: TableName)フィー ルドにそのテーブルのSQL名が表示されます。表示されない場合は、計算式 フィールドとCrystal Report内の選択計算式をチェックしてください。

# レポートを印刷する

#### 詳細レポート

#### 💋 注意:

詳細レポートの内容と、Crystal Reportsで詳細レポートを作成する方法については、「Crystal Reports」の章の「詳細レポートを作成する」の節を参照してください。

AssetCenterで詳細レポートを印刷するには、

- 1 詳細レポートの印刷が必要なレコードのリストを表示します(**[ポートフォ** リオ/資産とロット]など)。
- 2 印刷するレコードを複数選択します。
- 3 [ファイル/印刷]メニューを選択します。
- 4 [タイプ]フィールドを[詳細レポート(Crystal Reports)]に設定します。
- 5 レポートを選択します。
- 6 [印刷]をクリックします。

これで、選択したレコードごとの詳細レポートが作成されます。

#### 規定レポート

#### 💋 注意:

リストとグラフのレポートを**「規定レポート」**と呼びます。

「詳細レポート」と違い、テーブルはレポートのコンテキストにはなりません。

AssetCenterで規定レポートを印刷するには、

- 1 [ファイル/印刷]メニューを選択します。
- 2 [タイプ]フィールドを[規定レポート(Crystal Reports)]に設定します。
- 3 レポートを選択します。
- 4 [印刷]をクリックします。
- リストまたはグラフのレポートが印刷されます。

# モジュールに関連するCrystalレポートの識別

あるモジュールに関連するCrystalレポートを識別するには、

- 1 AssetCenterを起動します。
- 2 レポートのリストを表示します([ツール/レポート機能/レポート])。
- 3 リスト内で右クリックします。
- 4 ポップアップメニューから[リストの設定]を選択します。
- 5 リストの列見出しにリンク[ドメイン] (Domain)を追加します。
- 6 [OK]をクリックします。
- 7 [ドメイン]列でリストを並べ替えます。
- 8 モジュールのレポートはドメインの名前で識別できます。
   例:/ポートフォリオ管理/IT/レポート/



本章では、AssetCenterでアクションを定義する方法について説明します。 アクションを作成するには、[ツール/アクション/編集]メニューを選択し ます。

アクションは、**[ツール/アクション]**メニューまたはツールバーのアクショ ンリストから実行できます。

# アクションの定義

アクションは、AssetCenterから直接実行できるプログラムを呼び出す操作です。 次の種類のアクションがあります。

- [実行可能ファイル]
- [DDE]
- ・ メッセージ
- [スクリプト]: AssetCenterデータベースのオブジェクトを変更します。
- ・ [ウィザード]
- [印刷]
- [導入]
- [アクション]

メニューのリストからアクションを選択できるようにするには、まずこれらの アクションを定義する必要があります。

#### 💋 注意:

機能ドメインにより、アクションのドメインを定義し、機能ごとに分類するこ とができます。

### 機能ドメイン

AssetCenterでは、ソフトウェアの機能を分類するドメインを定義できます。数 種の機能ドメインがデフォルトで定義されています。これらのデフォルトの機 能ドメインは、[ファイル/モジュールの起動]メニューで起動または終了で きるモジュールに当たります。

機能ドメインは、[機能とお気に入り]枠内に表示される情報を作成し分類す るために使用されます。アクション用に機能ドメインを選択すると、アクショ ンは、機能とお気に入りの枠内の機能ドメインの名前の下に表示されます。

#### 💋 注意:

機能とお気に入りの枠の内容は状況(コンテキスト)に応じて変わります。ア クションが状況依存タイプである場合(例えば、従業員の画面が開いていない とアクションが実行されない、など)、現在のコンテキストがアクションのコ ンテキスト(この例では従業員のテーブル)に一致する時のみ、アクションが 表示されます。

機能ドメインを定義するには、次の手順に従います。

- 1 [新規作成]をクリックします。
- 2 [管理/機能ドメイン]メニューを選択します。
- 3 [名前]フィールドに機能ドメインの名前を入力します。この名前は機能と お気に入りの枠内に表示されます。AssetCenterは、機能ドメインにSQL名を デフォルトで割り当てます。このデフォルト値は変更可能です。
- 4 場合によっては機能ドメインの[**親ドメイン**]を選択します。
- 5 [作成]をクリックして作成を確定します。

### アクションを作成する

ここでは、アクションを作成する方法について説明します。

- アクションのタイプ
- 一般的な作成方法

AssetCenterの高度な使い方

- [DDE] タブページに入力する
- [**メッセージ**]タブページに入力する

### アクションのタイプ

AssetCenterでは、様々タイプのアクションを定義できます。

#### 💋 注意:

AssetCenterを使うと、[実行可能ファイル]、[メッセージ]、または[印刷] タイプのアクションを作成できます。[DDE]、[スクリプト]、および[ウィ ザード]タイプのアクションの場合は、既定のアクションを実行することしか できません。

### [実行可能ファイル]アクション

[実行可能ファイル]アクションは、プログラムを実行するアクションです。 このアクションによって、「.exe」、「.com」、「.bat」、「.pif」形式のアプリ ケーションを起動できます。その他の文書の場合も、拡張子がファイルマネー ジャ内のアプリケーションに関連付けられていれば開くことができます。

### [DDE]**アクション**

[DDE]アクションは、DDE要求を処理できるDDEサーバアプリケーション(またはDDE準拠のアプリケーション)にDDE要求を送ります。

DDEは、Dynamic Data Exchange (動的データ交換)の略で、プログラム間で動 的に情報を交換する手法です。AssetCenterではDDEを使って別のアプリケーショ ンでコマンドを実行できます。

例:DDEを使って、特定のファイルをMicrosoft Wordで開くように指定できます。

### [メッセージ]アクション

[**メッセージ**]アクションを使うと、次のメッセージシステムを経由してメッ セージを送信できます。

- AssetCenterの内部メッセージシステム
- 外部のVIM標準メッセージシステム(Lotus Notes、Lotus cc:Mailなど)
- 外部のMAPI標準メッセージシステム(Microsoft Exchange、Microsoft Outlook など)
- 外部のSMTP標準メッセージシステム

#### 🧶 警告:

ユーザが接続できるメッセージシステムでのみメッセージを送信できます。

VIM、MAPIまたはSMTP標準のメッセージを送信するには、AssetCenterで次の 情報を指定する必要があります。

- AssetCenterデータベースを開いた従業員([部署と従業員]テーブル)の詳細画面の、[メッセージ]タブページ/[アカウント](SQL名: MailLogin) フィールドおよび[パスワード](SQL名: MailPassword)フィールドで、 メッセージの送信者を指定します。
- 従業員の詳細画面の[全般]タブページ/[電子メール](SQL名: EMail) フィールドで、メッセージを受け取るためのアドレスを指定します。

AssetCenterの内部メッセージシステム経由でメッセージを送信するには、送信 者と受信者の両方の詳細画面の[プロファイル]タブページで、[ログイン] フィールドと[パスワード]フィールドを指定します。

#### 💋 注意:

AssetCenterユーザの内部メッセージ用アドレスは、[**ログイン**]と同じです。

#### 🐓 警告:

管理者は、名前が「Admin」のユーザを作成してから[アカウント]、[電子 メール]、[パスワード]の各フィールドに入力し、外部メッセージシステム を使ってAssetCenter Serverが正しく機能することを確認してください。

# [スクリプト]アクション

[スクリプト]アクションを使うと、AssetCenterデータベースのあらゆる操作 を実行できます。上級ユーザの操作範囲が広がり、他のタイプのアクションで は実行できない操作を行うことができます。特に次の操作を実行できます。

- レコードの作成
- レコードの削除
- レコードの複製
- AssetCenterデータベース内のオブジェクト(テーブル内のすべてのレコード、フィールド、リンクなど)の変更

このタイプのアクションで実行する操作は、Basicスクリプトで記述します。そのため、AssetCenter APIで使うような複雑な関数を使用できます。

#### 💋 注意:

[スクリプト]アクションで使用できる複雑な関数は、データベースの構造自体も変更できるため、この種のアクションを使用するとデータベースの整合性を損なう危険性があります。このため、スクリプトアクションの使用は適切な知識を持つ上級ユーザのみに限られます。

アクションのコンテキストに応じてデータベースオブジェクトの値を変更する には、様々な関数を使います。

- アクションにコンテキストがない場合は、AmSetFieldStrValue()または AmSetFieldLongValue()など、AssetCenter APIから抽出した関数を使う必要が あります。
- アクションのコンテキストとしてテーブルを指定した場合は、Set()関数を使うことができます。シンタックスは次の通りです。

Set [<Link.Link.Field>]=<値>

### [ウィザード]アクション

[ウィザード]は、繰り返し行われる複雑な作業を、AssetCenterで表示される ガイドに従って段階的に実行できるアクションです。ウィザードは、専用のプ ログラミング言語を使って設計します。

💋 注意:

ウィザードは複雑なアクションです。ウィザードについては、本マニュアルの 「**ウィザード」**の章を参照してください。

### [印刷]アクション

[印刷]アクションを使うと、レポートや書式を印刷できます。 この種のアクションでは以下のフィールドに値を入力します。

- **[タイプ]**(SQL名:seFormType)フィールド:印刷する文書タイプ(レポートまたは書式)を指定します。
- [レポート]または[書式]フィールド:どのレポートまたは書式を使用するのかを指定します。

#### 💋 注意:

アクションのコンテキストは、レポートまたは書式に定義されているコンテキ ストにより決定されます。

### [導入]アクション

[導入]アクションでは、コンピュータのテーブル内で定義されているコン ピュータのリスト上に、導入を起動できます。

このアクションは、コンピュータのテーブル内の状況依存アクションです。

この種のアクションでは以下のフィールドに値を入力します。

- 導入ワークフロー:作成またはインポートした導入ワークフローを選択します。
- 導入サーバ:導入のアプリケーションサーバを選択します。

導入の実施や設定の詳細については、マニュアル『Desktop Administration』を参照してください。

### [アクション]タイプのアクション

[**アクション**]タイプのアクションでは、指定されたクエリに応じて別のアクションを実行できます。

この種のアクションでは以下のフィールドに値を入力します。

- [実行するアクション]:起動するアクションを指定します。
- [選択クエリ]: アクションがどのコンテキストで実行されるのかを指定す るクエリのスクリプトを入力します。

#### 一般的な作成方法

アクションを作成するには、次の手順に従います。

- 1 [ツール/アクション/編集]メニューを選択します。
- 2 [新規作成]をクリックします。
- 3 アクションの名前を入力します。
- 4 **[タイプ]** (SQL名: seActionType)フィールドに、作成するアクションのタ イプを指定します。選択したアクションのタイプによって、次のタブページ が表示されます。
  - [実行可能ファイル]
  - [DDE]
  - [メッセージ]
  - [スクリプト]
  - [ウィザード]
  - [印刷]
- 5 必要に応じて、アクションの詳細画面の[SQL名](SQL名:SQLName) フィールドに入力します。SQL名からアクションを特定できます。特にDDE

コマンドを使ってアクションを実行する場合(AssetCenterをDDEコマンド サーバとして使う場合)に使います。

💋 注意:

[SQL**名]**フィールドに入力しないと、AssetCenterが自動的に標準的なSQL 名を生成して入力します。

- 6 次のようにして、[コンテキスト](SQL名:ContextTable)フィールドに 入力します。
  - ドロップダウンリストからテーブルを選択した場合は、アクションはそのテーブルに依存します。そのテーブルのレコードのリストまたはレコードの詳細画面を表示した場合にのみアクションが実行されます。
  - アクションにテーブルを指定しない場合は、ドロップダウンリストの一番上の[(テーブルなし)]オプションを選択します。
- 7 必要に応じて、[ドメイン]フィールドでアクションの所属先の機能ドメインを指定します。アクションは、機能とお気に入りの枠内のこのドメイン名の下に表示されます。
- 8 必要に応じて、アクションにアイコンを付けます。 アイコンを、アクションの詳細画面の左上に表示される四角い領域に挿入します。ここで指定したアイコンの画像は、ツールバーのアクションリストに表示されます。アクションリストから選択してアクションを実行すると、そのアクションのアイコンがツールバーに表示されます。
- 9 [説明]タブページのフィールドと、作成するアクションのタイプに固有の タブページのフィールドに値を入力します。
- 10 [作成]をクリックします。

💋 注意:

AssetCenterの管理者はすべてのアクションに目を通し、そのアクションが共有 されているかどうか、また管理者が作成したアクションかどうかを確認する必 要があります。

# [DDE] タブページに入力する

[DDE]アクションに関する情報は、アクションの詳細画面の[DDE]タブページに入力します。

このタブページは、アクションの**[タイプ]**(SQL名:seActionType)フィール ドを[DDE]に設定した場合にのみ表示されます。

DDE機能は、ソフトウェアに内蔵されているサービス機能を使います。DDE機 能を実行するには、コマンドの実行対象となるトピックを定義する必要があり ます。

次のように指定します。

- [サービス](SQL名: DDEService)フィールドに、呼び出す実行可能ファ イルのDDEサービスの名前を指定します。通常は、実行可能プログラム固有 のサービス名があります。サービスの一覧は、実行可能ファイルのマニュア ルを参照してください。
- [トピック](SQL名: DDETopic)フィールドに、アクションの実行対象と なるコンテキストを指定します
- [コマンド](SQL名:DDECommand)フィールドに、外部のアプリケー ションが実行するコマンドを指定します。
   Wordの場合は、WordBasicまたはVisual Basicコマンドになります。
   呼び出すアプリケーションのDDEサービスで複数のコマンドを使用できる場合は、複数のコマンドを並べて記述できます。
   この場合は、外部アプリケーションのシンタックスに従う必要があります。
- サービスが存在しない場合は、[ファイル](SQL名:ActionFile)フィールドに、サービスをアクティブにするアプリケーションを起動するファイルを指定します。これが、DDEコマンドに応答するメインアプリケーションになります。

#### 重要

外部アプリケーションに送信するコマンドは、大括弧([", "])で囲みます。 Microsoft Wordを使う例は以下の通りです。

[FileOpen("c: *l*tmp *l*test.txt")]

アクションのコンテキストを指定する場合は、変数を使ってデータベースのフィールド値を参照します。この場合は、変数も大括弧で囲むので、AssetCenterがコマンドと変数を区別できなくなります。区別できるようにするには、コマンドの大括弧の前にバックスラッシュ(/)を付けます。前述の例を使うと、次のように記述してアクションのコンテキストを指定できます。

/[FileOpen("c: /tmp /test.txt") /]

また、次のようにコマンドと変数を組み合わせることもできます。この例では[資産]テーブルをコンテキストとして指定します。

//FileOpen("c: /tmp /'+"[AssetTag]"+".txt") /] /[FileClose() /] /[FileExit() /]

アクションのコンテキストを指定しない場合は、バックスラッシュを使う必要はありません。大括弧で囲んだテキストは、外部アプリケーションに送信するコマンドと見なされます。

### [メッセージ]タブページに入力する

[メッセージ]アクションに関する情報は、アクションの詳細画面の[メッセージ]タブページに入力します。

このタブページは、アクションの**[タイプ]**(SQL名:seActionType)フィール ドを**[メッセージ]**に設定した場合にのみ表示されます。

#### 🌻 警告:

システムを正しく機能させるためには、システムのPATH変数にVIM DLL (VIM32.DLL)とMAPIDLL(MAPI32.DLL)を含んだフォルダを指定する必要 があります。

### [参照オブジェクト] (SQL名: RefObject) フィール ドの使用目的

このフィールドは、**[コンテキスト]**フィールドで選択したテーブルのリンク を選択するのに使います。

このフィールドは、AssetCenterの内部メッセージシステムで送信するメッセージ以外には使えません。ここでリンクを指定しておくと、メッセージの詳細画面で[参照オブジェクト]ボタンをクリックしただけで、メッセージの発信をトリガしたオブジェクトに直接アクセスできます。参照先のオブジェクトがアクションをトリガするレコードそのものの場合は、[参照オブジェクト](SQL名:RefObject)フィールドに入力する必要はありません。

### 配信済通知を受信する方法

メッセージが相手先に配信されたことを通知する配信済通知を、送信者が通常のメッセージサービス経由で受信するようにするには、[配信済通知](SQL 名:bAcknowledgment)チェックボックスをオンにします。

この通知は、AssetCenterデータベースを開いた従業員([部署と従業員]テー ブル)の[全般]タブページ/[電子メール](SQL名: EMail)フィールドに 指定したアドレスに送信されます。

#### 💋 注意:

AssetCenterの内部メッセージシステムや、MAPIまたはSMTPメッセージシステムで送信したメッセージの場合は、配信済通知を受信できません。

### アドレスを指定する

アドレスを指定する方法は以下の通りです。

### <メッセージエンジン>:<メッセージアドレス>形式のアド レス

<メッセージエンジン>には次の標準を指定できます。

- AM: AssetCenterの内部メッセージシステムを使用します。
- MAPI: MAPI標準のメッセージシステム(Internet Mail、Microsoft Outlookなど)を使用します。
- VIM: VIM標準のメッセージシステム (Lotus Notesなど)を使用します。
- SMTP: SMTP標準のメッセージシステム(インターネット標準)を使用します。

<メッセージアドレス>は、各メッセージシステムの通常の書式で指定します。 内部メッセージのアドレスは、ログイン名と同じです。

アドレスの例は以下の通りです。

- AM:Admin
- MAPI:CathyBernard@taltek.com
- VIM:Cathy Bernard / TALTEK
- SMTP:cbernard@taltek.com

### <AssetCenterログイン>形式のアドレス

従業員の[**ログイン**](従業員の詳細画面の[**プロファイル**]タブページ)が アドレス内で指定されている場合、この従業員の詳細画面/[**全般**]タブペー ジ/[**電子メール**](SQL名:EMail)フィールドに指定されているメッセージ システムが使用されます。

[電子メール]フィールドに値が入力されていないと、内部メッセージシステム経由でメッセージが送信されます。

例

- 次の3名のAssetCenterログインにメッセージを送信するとします: Cathy、 Gerald、Philip。
- 2 [電子メール]フィールドは、それぞれ次のように指定されています。Cathyの場合は「MAPI:CathyBernard@taltek.com」、Geraldは「VIM:Gerald Colombo / Taltek」、Philipの[電子メール]フィールドは空です。
- 3 送信者がMAPIアカウントを持っている場合、CathyにはMAPIで、他の2人に はAssetCenterの内部メッセージシステムでメッセージが送信されます。
- 4 送信者がVIMアカウントを有する場合、GeraldにはVIMで、他の2人には AssetCenterの内部メッセージシステムでメッセージが送信されます。

### 変数を使ったアドレス

アクションの対象がテーブルに依存する場合は、大括弧([])の中に変数を指定します。これらの変数は、AssetCenterデータベース内のフィールド値を参照します。

例: [資産] テーブルで選択した資産のユーザにメッセージを送信する場合は、 [User.Email]を使うことができます。

# アクションの例

ここでは、AssetCenterによるアクションの実行例を紹介します。

- [実行可能ファイル]タイプのアクション例
- [DDE]タイプのアクション例
- [メッセージ]タイプのアクション例
- [スクリプト]タイプのアクション例

### [実行可能ファイル]タイプのアクション例

次の画面はコンテキストを使わないアクションを指定した例です。このアクションは、AssetCenter Serverを起動してデータベース記述ファイルの「Acdemo」に 接続します。

#### 図 6.1. [実行可能ファイル]タイプのアクション

幅 アクションリモートコントロール'の詳細	-OX
名前: ノモートコントロール	新規作成( <u>N</u> )
コンテキスト: ()(テーブルなし)	複製(∐)
タイプ: 実行可能ファイル ・ ブレビュー(P)	
🥸 説明 👩 実行可能ファイルまたはDDE 任意管理項目 履. 👞	
「実行可能ファイルまたはDDEサービスの起動パラメーター	
ファイル: iftman.exe q	
7#ルダ:	
パラメータ: -host:[Name] -type:srv -mode:gui -close A C	
	変更( <u>M</u> )
	閉じる(©)

# [DDE] タイプのアクション例

[DDE]タイプのアクションを使うと、次のように様々な作業を実行できます。

- AssetCenterのデータをMicrosoft Excelのワークシートに挿入する。
- 発注に関連する情報を会計ソフトウェアに挿入する。
- チケットのクローズまたは作成を確認するメッセージを、ファックスで自動 的に送信する。
- 作業指示の依頼をファックスで自動的に送信する。
- ...
- ここでは、簡単な[DDE]アクションについて説明します。

### アクションの目的

このアクション例では、購入依頼の確認を送信します。

このアクションは、購入依頼の詳細画面からトリガされます。

このアクションでは、Microsoft Word7とAssetCenterとのDDEリンクを使います。 依頼に関する情報(契約担当者の詳細と依頼番号)をWord文書に挿入し、この 文書を印刷します。

## 準備:Wordでレターを作成する

まず、印刷するWord文書ファイル「LetterType.doc」を作成する必要がありま す。

このレターの構成は次の通りです。

#### 🛛 6.2. 「LetterType.Doc」

I	TALTEK SERVICES San Mateo Site	
<mrmrs> <firstname> <name> <adrt> <adr2> <zip> <city></city></zip></adr2></adrt></name></firstname></mrmrs>	IT Department	
	San Mateo. { DATE }	
Dear ≺t⁄mrMrs>,		
Following our conversation, I am pleased to confirm that your request has been attributed the following request number: <reqno>.</reqno>		
Regards,		

Procurement Manager

「LetterType.doc」文書を、AssetCenterのインストール先フォルダに格納します。 対応するWordテンプレート「Normal.dot」を「LetterType.doc」にリンクします。 このテンプレートには、次に示すマクロ「mymacro.bas」が含まれています。

```
Attribute VB_Name = "MyMacro"

Sub PrintLetterType(MrMrs, FirstName, Name, Adr1, Adr2, Zip, City, ReqNo)

'PrintLettreType Macro

Application.WindowState = wdWindowStateMinimize 'Run Winword in the back en

Documents.Open ("LetterType.doc") 'Open letter pattern

Documents("LettreType.doc").Activate

Selection.Find.ClearFormatting 'Clear parameters for Find function

Selection.Find.Replacement.ClearFormatting 'Clear parameters for Replace function

With Selection.Find

.Text = "<MrMrs>"

.Replacement.Text = MrMrs

.Forward = True
```

```
.Wrap = wdFindContinue
  .Format = False
  .MatchCase = False
  .MatchWholeWord = False
  .MatchWildcards = False
  .MatchSoundsLike = False
  .MatchAllWordForms = False
End With
Selection.Find.Execute Replace:=wdReplaceAll 'Execute replacement
With Selection.Find
  .Text = "<FirstName>"
  .Replacement.Text = FirstName
End With
Selection.Find.Execute Replace:=wdReplaceAll
With Selection.Find
  .Text = "<Name>"
  .Replacement.Text = Name
End With
Selection.Find.Execute Replace:=wdReplaceAll
With Selection.Find
  .Text = "<Adr1>"
  .Replacement.Text = Adr1
End With
Selection.Find.Execute Replace:=wdReplaceAll
With Selection.Find
  .Text = "<Adr2>"
  .Replacement.Text = Adr2
End With
Selection.Find.Execute Replace:=wdReplaceAll
With Selection.Find
  .Text = "<Zip>"
  .Replacement.Text = Zip
End With
Selection.Find.Execute Replace:=wdReplaceAll
With Selection.Find
  .Text = "<City>"
  .Replacement.Text = City
```

End With Selection.Find.Execute Replace:=wdReplaceAII

With Selection.Find .Text = "<ReqNo>" .Replacement.Text = ReqNo End With Selection.Find.Execute Replace:=wdReplaceAII

ActiveDocument.PrintOut 'Print document ActiveWindow.Close (wdDoNotSaveChanges) 'Close document w/o updating MsgBox ("Your document is being printed.") 'Notify user End Sub

### 手順1:AssetCenterでアクションを作成する

AssetCenterでアクションを作成するには、次の手順に従います。

- 1 [ツール/アクション/編集]メニューを選択して、アクションのテーブル を開きます。
- 2 [新規作成]をクリックし、アクションを作成します。
- 3 このアクションで表示する[購入依頼]テーブルを[コンテキスト](SQL 名:ContextTable)フィールドに指定します。
- 4 **[タイプ]**フィールドを [DDE]に設定します。
- 5 [DDE] タブページで、AssetCenterとMicrosoft Word7とのDDEリンクを定義 します。

DDEタブページの各フィールドに次のように指定します。

- 1 **[サービス]** (SQL名: DDEService) フィールドを「Winword」に設定します。
- 2 [トピック] (SQL名: DDETopic) フィールドを「System」に設定します。
- 3 [サービス開始パラメータ]セクションでは、Winword.exeとそのパスを指定します。
- 4 [コマンド] (SQL名: DDECommand) フィールドには、次のように起動す るマクロとそのパラメータを指定します。

/[MyMacro.ImprimeLettreType "[Requester.MrMrs]", "[Requester.FirstName]", "[Requester.Name]", "[Requester.Location.Address1]", "[Requester.Location.Address2]", "[Requester.Location.ZIP]", "[Requester.Location.City]", "[ReqNumber]" /]

[作成]をクリックし、アクションの作成を確定します。

### 手順2:アクションを起動する

アクションを起動するには、次の手順に従います。

AssetCenter

- 1 [購入依頼]テーブルを開きます。
- 2 購入依頼を選択します。
- 3 [ツール/アクション]メニューからアクションを起動します。

アクションを起動すると、次の作業が行われます。

- 1 Microsoft Wordが起動し、「LetterType.doc」が読み込まれます。
- 2 連絡担当者の詳細と依頼番号がレターに挿入されます。
- 3 レターが印刷されます。

# [メッセージ]タイプのアクション例

この例では、資産のリストから、ある資産の使用者に資産のリース期間が満了 間近であることを知らせるメッセージを送信します。この資産は、取得方法が [リース]であり、リース明細にリンクされている必要があります([取得] タブページ)。参照するオブジェクトをリース明細にするには、アクションを 次のように設定します。

#### 図 6.3. 参照するオブジェクトを指定するメッセージタイプのアクション の設定例



# [スクリプト]タイプのアクション例

[スクリプト]タイプのアクションの作成では、AssetCenterデータベースを変更するBasicスクリプトを記述する必要があります。

#### 💋 注意:

スクリプトタイプのアクションで使用できる関数については、マニュアル**『プ** ログラマーズリファレンス』、「関数の用途別一覧」の章、「組み込み関数」 の節を参照してください。

#### 準備

アクションの作成を準備する手順は以下の通りです。

- 1 [ツール/アクション/編集]メニューを選択し、アクションの詳細画面で [新規作成]ボタンをクリックします。
- 2 作成するアクションに「テスト」などの名前を割り当て、[タイプ](SQL 名:seActionType)フィールドを[スクリプト]に設定します。アクション のコンテキストは選択しないでください。[作成]をクリックします。
- 3 [スクリプト]タブページで国ボタンをクリックし、Basicスクリプトのビルダ画面を表示します。スクリプトタイプのアクションで使用可能なSuccess() 関数には、明示的な戻りコードは不要です。以下の例では、次の表に示す情報に基づいて属性のテーブル内に新しいレコードを作成します。

フィールドのラベル	フィールドのSQL名	フィールドの値
[名前]	Name	PC
[作成]	seBasis	ポートフォリオ品目
[接続可能]	blsCnxClient	このチェックボックスはオン
		になっています。

### スクリプトを記述する

次のように入力します。

Dim Irec As Long Dim Ires As Long Irec=AmCreateRecord("amNature") Ires=AmSetFieldStrValue(Irec, "Name", "PC") Ires=AmSetFieldStrValue(Irec, "seBasis", 1) Ires=AmSetFieldStrValue(Irec, "bIsCnxClient", 1) AmInsertRecord(Irec)

#### 💋 注意:

このアクションにより、ユーザの介入なしに希望の属性を作成できるようになります。

### Set() 関数の使用例

ここでは、コンテキストに**[属性]**テーブルを指定して、**[スクリプト]**タイプのアクションで前の例と同じ属性を作成します。次のようにスクリプトを記述します。

Set [Name]="PC" Set [seBasis]=1 Set [bIsCnxClient]=1

💋 注意:

このアクションを実行するには、ユーザが**[属性]**テーブルを開き、**[新規作 成]**をクリックする必要があります。また、スクリプトを実行した後に、**[作 成]**をクリックして作成を確定する必要があります。

### ヒント

スクリプト内の1つのアクションを無効にする場合は、戻りコードの値を0以外 (例えば12001)に設定します。この値はエラーコードと見なされます。次のコ マンドはアクションを中断し、すべての変更を取り消します。

RetVal=12001

### 変数を使う

状況依存アクションの詳細画面の**[実行可能ファイル]**、**[**DDE**]**、および **[メッセージ]**タブページでは、データベース内のフィールド、任意管理項目 または特殊フィールドの値を参照する変数を使うことができます。

変数の書式は [Link.Link.Field]です。

入力するフィールドの右側にある 図をクリックすると、特殊文字列画面が表示 され、より簡単に変数を入力できます。

大括弧([])の外にあるものはすべてテキストと見なされます。

例: [Link.Link.Field].docは、Link.Linkというリンクを介してメインテーブルにリ ンクしたテーブル内のFieldの値を呼び出します。

#### 🌻 警告:

アクションで変数を使う場合は、アクションの詳細画面の**[コンテキスト]** フィールドでAssetCenterのテーブルを指定し、アクションを実行する前にその テーブルのレコードリストでレコードを選択しておく必要があります。
## Sybase SQL Anywhereデータベースエンジンを使う 場合

Sybase SQL AnywhereをAssetCenterのデータベースエンジンとして使う場合は、 アクションのフィールドの先頭に「 $\{d$ 」または「m」を記述することはできません。

アクションのフィールドの先頭に「{d」または「m」を記述する必要がある場合は、文字列の前にスペースを1つ挿入してください。

## アクションをテストする

アクションを作成時にテストするには、アクションの詳細画面の右上隅に表示 される[プレビュー]ボタンを使います。

## [計算] ボタン

コンテキストを選択し、[計算]ボタンをクリックします。これで[実行可能 ファイル]、[DDE]、および[メッセージ]のいずれかのタブページのフィー ルドに値が入ります。[コンテキスト](SQL名:ContextTable)フィールドで 選択したレコードから、変数が正しく抽出されていることを確認します。

#### [実行]ボタン

このボタンをクリックすると、プレビュー画面から直ちにアクションを実行できます。

## アクションを実行する

次のいずれかの方法でアクションを実行できます。

- 🕐 ツールバーのドロップダウンリスト <sup>%</sup> を使います。
  - % ボタンは、使用しているコンピュータ上でアクションを実行するたびに、そのアクションのボタンで置き換えられます。アクションを既に実行している場合は、% アイコンまたは置き換えられたアイコンをクリックすると、アクションのアイコンがリセットされます。

- ボタンをクリックすると、使用可能なアクションのリストが表示されます。
- このドロップダウンリストをツールバーに挿入するには、[ツール/ツー ルバーのカスタマイズ]メニューを使います。このメニューは「ツール」 カテゴリにあります。
- [ツール/アクション]メニューを使います。サブメニューから希望のアクションを選択します。
- アクションの詳細画面の右上隅にある[プレビュー]ボタンを使います。
  - 実行するアクションにコンテキストを指定する必要がある場合は、【コンテキスト】(SQL名: ContextTable)フィールドにアクションの参照テーブルのレコードを指定します。
  - [実行]をクリックして、アクションを実行します。
- ポップアップメニュー(マウスの右ボタンをクリックして表示)を使います。開いているテーブルで使用できるアクションがある場合は、ポップアップメニューにアクション名が表示されます。

#### リストで複数のレコードを選択

リストから選択した複数のレコードに1つのアクションを適用できます。 例えば複数の資産を選択し、そのユーザに同じメッセージを送信することが可 能です。

## 「ウィザード」タイプのアクション

ウィザードは連続するページで構成されています。ユーザは、各ページで情報 を選択したり、データを入力したりします。

ウィザードでは、次のように簡単にページ間を移動できます。

- ページに情報を正しく入力したら、[次へ]ボタンをクリックして次のページに移動できます(移動先のページはトランジションの設定によります)。
   最後のページまで進むと、このボタンは使用できなくなります。
- [戻る]ボタンをクリックすると、いつでも前のページに戻り、情報を訂正 することができます。
- [終了]ボタンをクリックすると、ウィザードの最終アクションをいつでも 実行できます。情報不足のためにウィザードが指定されたタスクを実行でき ない場合は、必要な情報を入力するためのページが表示されます。

[**キャンセル**]ボタンをクリックすると、ウィザードの実行(および関連する アクション)を完全に取り消すことができます。



AssetCenterでは、次の2種類のメッセージを管理できます。

- AssetCenterから発信され、内部メッセージシステムを介してAssetCenterデー タベースに送信されたメッセージ
- AssetCenterで作成され、外部メッセージシステムを介して送信されたメッ セージ

## メッセージシステムの仕組み

AssetCenterは以下のタイプのプロトコルを使用してメッセージの送信を管理します。

- AM (AssetCenter)
- SMTP
- MAPI
- VIM

AssetCenterは、受信時にはAM (AssetCenter) タイプのメッセージのみを管理します。

#### 図 7.1. メッセージシステムの仕組み



## メッセージを送信する方法

メッセージは、**[メッセージ]**タイプのアクションを介して作成されます。 メッセージを送信するには、まずメッセージを送信するアクションを作成する 必要があります。

メッセージを送信するアクションは、次の方法でトリガされます。

- [ツール/アクション]メニューに表示されるリストからアクションを選択して、手動でトリガする。
- AssetCenter Serverによって自動的にトリガされる。
- AssetCenterによって自動的にトリガされる。

[メッセージ]タイプのアクションの作成については、本マニュアルの「アク ション」の章、「アクションを作成する/[メッセージ]タブページに入力す る」の節を参照してください。

## メッセージを参照する方法

#### 内部メッセージシステムで受信したメッセージを参照す る

AssetCenterエージェントは新しいメッセージが到着しているかどうかを確認し、 ユーザに新着メッセージの受信を通知します。

受け取ったメッセージは次の方法で参照できます。

[ツール / メッセージ]メニューを使う。

• 新着メッセージを通知するメッセージボックスで参照する。

#### 外部メッセージシステムで受信したメッセージを参照す る

メッセージの受信者は、外部メッセージシステムのクライアントプログラム上 でメッセージを参照できます。

#### 配信済通知

#### 💋 注意:

AssetCenterの内部メッセージシステムや、MAPIまたはSMTPメッセージシステムで送信したメッセージの場合は、配信済通知を受信できません。

配信済通知を受信するには、**[メッセージ]**タイプのアクションの詳細画面で **[配信済通知]**(SQL名:bAcknowledgment)チェックボックスをオンにしま す。

この通知は、AssetCenterデータベースを開いた従業員([部署と従業員]テー ブル)の[全般]タブページ/[電子メール](SQL名: EMail)フィールドに 指定したアドレスに送信されます。



本章では、ワークフローチャートを定義し、管理する方法について説明します。

定義

ここでは、ワークフローで使われるいくつかの主要な用語について説明します。

- ワークフロー
- ワークフローのアクティビティ
- ワークフローのイベント
- ワークフローのトランジション
- ワークフローのタスク
- ワークフローのアクティビティの担当者
- ワークフローの実行グループの定義

#### ワークフロー

ワークフローは、業務処理を標準化し、自動化する機能です。 例えばワークフローを使うと、次のようなプロセスのモデルを作成し、プロセ スを自動的に処理することができます。

- 購入依頼の承認手続き
- 資産の移動
- その他

AssetCenterを使うと、ワークフローチャートを定義し、その進捗状況を管理することができます。

#### ワークフローチャート

AssetCenterでワークフローチャートを作成するには、次の要素を定義します。

- アクティビティ
- トリガイベント(アクティビティの結果発生したイベントで、トランジションを起動するイベント)
- トリガトランジション(アクティビティを起動するトランジション)
- コンテキスト
- 制限時間とアラーム

図 8.1. ワークフローチャートの概略



ワークフローチャート (SQL名:mWfScheme)のテーブルにアクセスするには、 [ツール/ワークフロー/ワークフローチャート]メニューを選択します。

#### ワークフローインスタンス

「ワークフローインスタンス」とは、実行中の定義済みワークフローチャート を指します。

## ワークフローのアクティビティ

ワークフローのアクティビティは次の要素で構成されます。

- 実行するタスク。このタスクにはユーザによる操作が必要なものと、 AssetCenter Serverで自動的に実行できるものがあります。
- 他のアクティビティへのトランジションを起動するイベント

ワークフローのアクティビティは、**[ワークフローのアクティビティ]**(SQL 名:amWfActivity)テーブルに格納されます。 ワークフローのアクティビティの詳細画面を表示するには、**「ツール/ワーク** 

**フロー/ワークフローチャート**]メニューを選択し、**[アクティビティ]**タブ ページのグラフィカルエディタを使います。

#### ワークフローのイベント

ワークフローのイベントは、アクティビティを実行した結果発生します。ワー クフローのイベントにより、別のアクティビティを引き起こすトランジション が起動されます。

ワークフローチャートに属するイベントは、ワークフローのイベント(SQL名: amWfEvent)のテーブルに格納されます。

これらのイベントが発生すると、ワークフローインスタンス(SQL名: amWfOccurEvent)の基本イベントのテーブルに記録されます。

ワークフローのイベントの詳細画面にアクセスするには、[ツール/ワークフ ロー/ワークフローチャート]メニューを選択し、[アクティビティ]タブペー ジにあるグラフィカルエディタを使います。

#### ワークフローのトランジション

ワークフローのトランジションにより、1つのアクティビティから別のアクティ ビティへ移行できます。トランジションは、イベントにより発生します。 1つのイベントを複数のトランジションに関連付けることができます。

ワークフローチャートに属するトランジションは、「**ワークフローのトランジ** 

**ション**] (SQL名: amWfTransition) テーブルに格納されます。

ワークフローのトランジションの詳細画面にアクセスするには、[ツール/ワー クフロー/ワークフローチャート]メニューを選択し、詳細画面の[アクティ ビティ]タブページにあるグラフィカルエディタを使います。

#### ワークフローのタスク

ワークフローのタスクは、実行するように割り当てられた業務で、アクティビ ティをトリガすると発生します。

ワークフローのタスクを**[ワークフローのタスク]**(SQL名:amWfWorkItem) テーブルに記録するには、アクティビティの詳細画面の**[全般]**タブページに ある**[タスクを保存]**(SQL名:bLogWorkItem)チェックボックスをオンにす る必要があります。

実行するワークフローのタスクのリストにアクセスするには、[ツール/進行 中のタスク]メニューを選択します。

#### ワークフローのアクティビティの担当者

担当者は、「質問」または「ユーザのアクション」タイプのワークフローアク ティビティから発生するタスクを実行する人です。担当者は、「自動アクショ ン」または「テスト/スクリプト」タイプのアクティビティには関わりません。 ワークフローの担当者は、[ワークフローの役割](SQL名:amWfOrgRole) テーブルに格納されます。ワークフローの役割のテーブルにアクセスするには、 [ツール/ワークフロー/役割]メニューを選択します。

#### ワークフローの実行グループの定義

ワークフローの実行グループを使って、定義しているワークフローチャートを 分類できます。ワークフローチャートの実行グループは、[ツール/ワークフ ロー/ワークフローチャート]メニューを選択し、[全般]タブページの[実 行グループ](SQL名: GroupName)フィールドで指定します。



AssetCenterでワークフローを作成するには、まず[ツール/ワークフロー/ワー クフローチャート]メニューを選択してグラフィカルエディタを表示し、ワー クフローチャートを定義します。ワークフローチャートでは、アクティビティ、 イベント、およびトランジションを定義します。これらの要素を、AssetCenter のアクションと従業員(ワークフローの担当者)に関連付けることができます。 ワークフローチャートは、ワークフローエンジンによって解釈されます。ワー クフローエンジンは、AssetCenter ServerまたはAssetCenterのエージェントで実行 されます。

ワークフローエンジンは、イベントに反応して、ワークフローをトリガしモニ タします。ワークフローエンジンは、次の内容を実行します。

- 実行するタスクを生成する。
- これらのタスクとアクティビティを起動するイベントをモニタする。
- 受信するイベントと実行するユーザタスクの履歴を作り、イベントの推移を 記録する。

ワークフローのタスクは、ワークフローエンジンまたはAssetCenterユーザによっ て実行されます。実行されると、さらにイベントが起動され、ワークフローエ ンジンにその情報が伝えられます。

AssetCenterでワークフローを実行する際の概要は、次の図の通りです。

図 8.2. AssetCenterのワークフロー - 概要



ワークフローインスタンスの処理は、アクティビティやイベントの属性、およ びこれらを制御するワークフローエンジンの設定によって異なります。

## ワークフローに関連するメインテーブル

ワークフローに関連するメインテーブルとこれらの関係は、次の図の通りです。 テーブルは、ラベルとSQL名で表記されています。



図 8.3. ワークフローチャートを定義できるメインテーブル

進行中のワークフローに関連するメインテーブルは、次の図の通りです。

#### 図 8.4. 実行中のワークフローに関連するメインテーブル



## ワークフローのグラフィカルエディタの使用方法

ワークフローチャートにアクセスするには、[ツール/ワークフロー/ワーク フローチャート]メニューを使います。このメニューを使用できるのはAssetCenter の管理者だけです。

ワークフローチャートの詳細画面の[**アクティビティ**]タブページでは、ワー クフローチャートを簡単に作成できるグラフィカルエディタを使用できます。 ここでは、このグラフィカルエディタを使って、ワークフローチャートを作成、 変更、または削除する方法について説明します。

- アクティビティ
- イベント
- トランジション
- その他の機能

#### アクティビティ

アクティビティを作成するには、次のいずれかの操作を行います。

- [アクティビティ]タブページの空白の領域をマウスの右ボタンでクリックして、ポップアップメニューから[アクティビティの追加]コマンドを選択します。アクティビティの詳細画面が表示されます。
- ボタンをクリックし、次にグラフィカルエディタ領域をクリックします。
   アクティビティの詳細画面が表示されます。

アクティビティを削除するには、次のいずれかの操作を行います。

- アクティビティをクリックして選択し、[Delete]キーを押します。
- アクティビティを選択し、ポップアップメニューから「削除」コマンドを選択します。
- アクティビティを選択し、 ボタンをクリックします。

アクティビティの詳細設定を変更するには、次のいずれかの操作を行います。

- アクティビティをマウスの右ボタンでクリックし、ポップアップメニューから[アクティビティの詳細]を選択します。
- アクティビティをダブルクリックします。

AssetCenter



#### イベント

## [データベース]または[定期的]タイプのイベント

アクティビティに出力イベントを追加するには、次のいずれかの操作を行いま す。

- アクティビティをマウスの右ボタンでクリックし、ポップアップメニューから[イベントの追加]コマンドを選択します。
- アクティビティを選択し、<sup>国</sup>ボタンをクリックします。

アクティビティから[データベース]または[定期的]タイプの出力イベントを削除するには、次のいずれかの操作を行います。

- イベントを選択し、[Delete]キーを押します。
- イベントをダブルクリックし、ポップアップメニューから[削除]コマンド を選択します。
- イベントを選択し、 ボタンをクリックします。

[**データベース**]または[**定期的**]タイプのイベントの詳細を変更するには、 次のいずれかの操作を行います。

- イベントをダブルクリックします。
- イベントを選択し、ポップアップメニューから[イベントの詳細]を選択します。

#### [システム]イベント

[**システム**]イベントは、各イベントの起点のアクティビティから作成および 削除できます。

[システム]イベントを変更するには、必要に応じて次のいずれかの方法を使います。

 イベントの処理方法(イベントの詳細画面の[処理](SQL名: seProcessingMode)フィールド)を変更するには、[データベース]または [定期的]タイプのイベントを変更するのと同じ方法をとります。

それ以外の場合は、イベントの起点であるアクティビティの詳細設定を編集し、変更します。

## トランジション

トランジションを作成するには、次の手順に従います。

- 1 開始イベントをクリックして選択します。
- 2 マウスボタンを押したまま、リンク先のターゲットアクティビティにドラッ グします。

トランジションを削除するには、次のいずれかの操作を行います。

- トランジションをクリックして選択し、 [Delete] キーを押します
- トランジションを選択し、ポップアップメニューから[削除]コマンドを選択します
- トランジションを選択し、 ボタンをクリックします。

トランジションの詳細設定を変更するには、次の手順に従います。

1 トランジションをクリックして選択します。

2 ポップアップメニューから[トランジションの詳細]を選択します。

トランジションのソースまたはターゲット、またはその両方を変更するには、 次の手順に従います。

- 1 トランジションを選択します。
- 2 変更するソースまたはターゲットをドラッグします。

#### その他の機能

グラフィカルエディタを使って、次の操作を行うこともできます。

- リンクしているアクティビティとトランジションのドラッグ&ドロップ
- [**ズーム**] スライダまたは 🭳 ボタンを使ったフローチャートの拡大と縮小

## ワークフローの活用法

AssetCenterのワークフローを使って、会社の業務処理を自動化することができます。手順は以下の通りです。

- 1 自動化する会社の業務を分析します。
- 2 次の要素を作成します。
  - 1 ワークフローの役割
  - 2 アクション

- 次の要素を定義してワークフローチャートを作成します。
   アクティビティ、イベント、およびトランジション
   アラーム
- 4 ワークフローの適切な実行グループを指定します。各ワークフローチャート をワークフローの実行グループに関連付けます。
- 5 1台または複数のマシンでAssetCenter Serverを起動します。各AssetCenter Server で、サーバがモニタするワークフローの実行グループと、モニタ用パラメー タを定義します。

## ワークフローを使った依頼承認手続き例

ここでは、簡単なワークフローの例について説明します。

- 1 目的
- 2 必要条件
- 3 アクティビティを作成する
- 4 アクティビティと同時に作成されるイベントのパラメータ
- 5 開始イベントを作成する
- 6 トランジションを作成する
- 7 ワークフローの起動例

目的

このワークフローチャートを使って、次の手順の購入依頼プロセスを自動化します。

#### 図 8.5. ワークフローチャート - 依頼の承認



ワークフローチャートの手順は次の通りです。

- 1 ワークフローインスタンスは、購入依頼が承認されるべき状況になると(つまり、購入依頼の[依頼ステータス](SQL名:seStatus)フィールドが[承認待ち]に設定された時)開始します。
- 2 まず責任者の許可をもらいます。この手順では、依頼を部署の責任者に提出して承認を得ます。依頼はメッセージで通知されます。承認依頼メッセージを発信した翌日までに承認者が依頼を処理しない場合は、催促のアラームをトリガするようにプログラムします。
- 3 責任者の承認をもらったら、財務承認手続きに移ります。 それ以外の場合は、依頼は却下されます。(3b)
- 4 財務承認では、依頼を会社の会計監査役である木村氏に提出します。ここで も依頼はメールで通知され、承認を催促するアラームも設定します。
- 5 会計監査役が購入依頼を許可すると、購入依頼は承認されます。 それ以外の場合は、依頼は却下されます。(5b)
- 6 購入依頼が承認されると、購入依頼の詳細画面の[依頼ステータス]フィー ルドが[承認済]に設定されます。 購入依頼が却下された場合、[依頼ステータス](SQL名:seStatus)フィー ルドは[却下]に設定されます。

#### 必要条件

「Admin」ログインを使ってデータベースに接続し、メッセージを設定する必要 があります。(詳細は本マニュアルの「メッセージ」の章を参照してくださ い。)

#### ワークフローの担当者を作成する

このワークフローチャートに関わる担当者は次の通りです。

- 依頼者の部署の責任者
- 会社の会計監査役、木村五郎氏

これらのワークフローの担当者は**[ワークフローの役割]**(SQL名: amWfOrgRole)テーブルに保存されます。これらの担当者を作成するには、 **[ツール/ワークフロー/役割]**メニューを選択して**[新規作成]**をクリック します。

## 依頼者の部署の責任者

この担当者はスクリプトによって計算されます。これを定義するには、役割の 詳細画面で次のように入力します。

■ 役割 '依頼者の部署の責任者' の詳細					
△役割		△コンテキ)	スト	+	新規作成(N)
依頼内容の承認		amRequest			
青任者の承認		am Request			(□) 類 数
開務の承認		am Request		-	削除(D)
		annicquest		+	
•			•		
		_ △ 🎨	5/5		
役割: 依頼者の部署の責任者	参照番号: D00100	1		=	
コンテキスト: 📰 依頼 (am Request) 🔹	タイプ: 特定の	個人に割当		•	
スクリプト:					
Retval=[Requester.Parent.Supervisor]			4	٩	
T			¥		閉じる(6)

#### 会計監查役

この担当者は木村五郎氏として指定します。役割の詳細画面で次のように入力します。

℡ 役割 '会計監査役' の詳細		<u>- 0 ×</u>
▲役割 토((井)) - 조진		新規作成( <u>N</u> )
育住者の承認 財務の承認	am Request	複製(∐)
依頼者の部署の責任者  会計監査役	am Request	削除( <u>D</u> )
•	+	
	△ 🔅 676	
役割: 会計監査役	参照番号: D001002	
コンテキスト: 🗊 (テーブルなし) 🔹	タイプ: 個人に割当 🔹	
担当者:木村五郎		
		閉じる( <u>C</u> )

#### アクションを作成する

ワークフローチャートでは、数多くのアクションを使います。これらを作成するには、[ツール/アクション/編集]メニューを選択します。

## 依頼者の部署の責任者に送られる技術検証依頼

このアクションは、技術検証段階で使われます。このアクションで、この依頼 を検討する必要があることを技術検証の担当者に通知できます。

🍕 アクション 技術検証	を頼'の詳細		
名前:	技術検証依頼		
コンテキスト:	■ワークフローのタスク (amWfWor	kItem) 🔹	
タイプ:	メッセージ	<ul> <li>プレビュー(₽)</li> </ul>	
🤥 説明 🖂	メッセージ 任意管理項目 履歴	₫ 書類	
参!	オブジェクト:	•	
	優先度: 普通	•	
	□ 配信済通知		
受信者: [Assig	nee.Email]	✓ × <	
Cc:		٩	
Bee:		٩	
件名:購入(	頼の検証	٩	
メッセージ: 購入	5頼の申込書の再確認をお願いしま	<b>す</b> 。 ▲ 🍳	
			変更( <u>M</u> )
		<b>_</b>	キャンセル

### 会計監査役に送られる財務検証依頼

「財務検証依頼」アクションは、財務承認プロセスで使われます。このアクションにより、この依頼の検討が必要であることを伝えるメッセージが、財務検証 担当者に送信されます。

🔍 アクション 財務	8検証依頼'の詳細	
	名前: 財務検証依頼	
コンテキ	-スト:  ゴワークフローのタスク (am WfWorkItem) ・	
Ś	イブ: メッセージ ・ プレビュー(P)	
🥸 説明	≥ メッセージ 任意管理項目 履歴 書類	
	参照オブジェクト:	
	優先度: 普通 ▼	
	配信済通知	
受信者:	[Assignee.EMail]	
Cc:	٩	
Bcc:	<u> </u>	
件名:	購入依頼の検証	
メッセージ:	購入依頼の申込書の再確認をお願いします。	作成©)
		作成(R) 👁
	×	キャンセル

#### 購入依頼の承認

このアクションは、後に作成する**「依頼の承認」**アクティビティの時点で使用 されます。

このアクションにより、購入依頼の詳細画面の**[依頼ステータス]**(SQL名: seStatus)フィールドが**[承認済]**に設定されます。このアクションは、**[スクリプト]**タイプのアクションです。

[依頼ステータス]フィールドは、システムリストデータです。値のリストを 表示するには、次の手順に従ってこのフィールドのヘルプを表示します。

- 1 依頼の詳細画面の**[依頼ステータス]**フィールドをマウスの右ボタンでク リックします。
- ポップアップメニューから[フィールドのヘルプ]メニューを選択します。
   [承認済]として表示される値は、データベースには「14」として保存されます。

AssetCenterの高度な使い方

アクションの詳細は次のようになります。



#### 購入依頼の却下

このアクションは、後に作成する「依頼の却下」アクティビティの時点で使用 されます。

「購入依頼の承認」アクションに似ていますが、購入依頼の詳細画面の[全般] タブページにある[依頼ステータス](SQL名:seStatus)フィールドは、[却 下]になります。

◎ アクション'購入依頼の却下'の詳細	
名前:購入依頼の却下	
コンテキスト: III 依頼 (am Request) ・	
タイブ: スクリプト ・ プレビュー(P)	
🥸 説明 🔂 スクリプト 任意管理項目 履歴 書類	
アクションのスクリプト:	
Set [seStatus] = 15	
	作成①
	作成(R) 👁
	キャンセル

## カレンダを作成する

カレンダのリストには、**[ツール/カレンダ]**メニューからアクセスできます。 このカレンダはワークフローチャートのアクティビティに関連付けられます。 これにより、ワークフローチャートの処理の締切りを設定できます。

「風カレンダ」標準カレ	ンダ-日本'の詳細		_ 🗆 X
名前: 相	黒準カレンダ - 日本		新規作成( <u>N</u> )
SQL名: ja	apanese_calendar		複製(∐)
タイムゾーン:		۹ 🗉 🔻	
予定表(I) 例 週間予定表の定義	外④   ブレビュー☆   髪		
月曜日( <u>M</u> ):		9:00-12:00;13:00-18:00	
火曜日( <u>T</u> ):		9:00-12:00;13:00-18:00	
水曜日(₩):		9:00-12:00;13:00-18:00	
木曜日( <u>R</u> ):		9:00-12:00;13:00-18:00	
金曜日( <u>F</u> ):		9:00-12:00;13:00-18:00	
土曜日(5):			
日曜日( <u>D</u> ):			
			閉じる( <u>C</u> )

#### ワークフローチャートを準備する

- 1 [ツール/ワークフロー/ワークフローチャート]メニューを選択します。
- 2 [新規作成]をクリックします。
- 3 ワークフローに「依頼の承認」という名前を付けます。
- 4 デフォルトで、ワークフローチャートのすべてのアクティビティに適用されるフローチャートの開始テーブルが表示されます。この例では、【依頼】 (SQL名: amRequest)のテーブルです。
- 5 [作成]をクリックします。開始アクティビティ(「開始」)は、[アク ティビティ]タブページのグラフィカルエディタで自動的に作成されます。

AssetCenter



## アクティビティを作成する

アクティビティは、ワークフローチャートの詳細画面の**[アクティビティ]**タ ブページ内で、グラフィックモードで作成できます。

- 1 ワークフローのオブジェクト以外の場所にカーソルを置きます。
- 2 右クリックします。
- 3 ポップアップメニューから[**アクティビティの追加**]メニューを選択しま す。アクティビティの詳細画面が表示されます。

#### 「技術検証」アクティビティを作成する

- 1 アクティビティに「技術検証」という名前を付けます。
- 2 部署の責任者に承認の依頼を提出するアクティビティなので、[タイプ] (SQL名:seType)フィールドのドロップダウンリストから[質問]という 値を選択します。
- 3 [全般]タブページの[コンテキスト](SQL名: ContextTable)フィール ドは変更しません。

#### 「技術検証」アクティビティのパラメータ 1 [パラメータ] タブページに次のように入力します。

🗟 ワークフローチャート '承認依頼' のアク:	ティᡛティ	
名前: 技術検証	テンプレート:	<. ■ -
タイプ: 質問	▼ 参照番号: A001	003
全般 バラメータ 🙆 時間 🧯	🖻 アラーム 🛛 履歴 🛛 書類	[]
担当者		
担当者: 依頼者の	の部署の責任者	
通知する: はい	<ul> <li>アクション:</li> </ul>	技術検証依頼 🔍 🖬 👻
愛問またけお子・		
購入依頼 [RegNumber] を承認してい	ただけますか?	- Q
		-
答の選択肢		
答: 承認		
SQL名:		
		変更( <u>M</u> )
▲認 却下		キャンセル

- 2 尋ねる質問事項を指定します。
  - 1 質問のテキストが購入依頼の番号を参照するように指定します。
  - 2 却下と承認の2つの回答を用意します。質問に対する答を説明するサブタ ブページを追加するには、サブタブのラベル部分をマウスの右ボタンで クリックし、[リンクレコードの追加]または[リンクレコードの複製] を選択します。
- 3 [担当者](SQL名: Assignee)フィールドに質問の宛先を指定します。この場合、ワークフローの担当者は依頼者の部署の責任者です。この担当者は、前の準備段階でワークフローの役割のテーブルに作成されています。
- 4 ワークフローの担当者にこの依頼を検討するよう通知するには、次のように 指定します。
  - 1 **[通知する]**(SQL名:bNotifAssignee)フィールドを**[はい]**に設定します。
  - 2 実行するアクションを指定します。これは、前の準備段階で作成した「技 術検証依頼」アクションです。このアクションは、購入依頼の技術検証 を開始すると、自動的にトリガされます。

#### 「技術検証」アクティビティの制限時間を指定する

アクティビティの詳細画面の**[制限時間]**タブページで次の内容を指定します。

- 1 アクティビティにリンクする業務用カレンダを指定します。このカレンダを 使って制限時間が計算されます。前の準備段階で作成した「日本カレンダ」 を選択します。
- 2 アクティビティがトリガされる時間を基準にして、決定を下さなければならない時間を定義します。この例では、ワークフローの担当者は翌日までに質問に答えることにします。

■ ワークフローチャート '承認	!依頼' のアクティビティ		
<mark>名前:</mark> 技術検証	テンプレート:	• ۵ •	新規作成( <u>N</u> )
タイプ: 質問	✓ 参照番号: A001003		複製(Ⅲ)
全般 バラメータ 🙆	時間 📸 アラーム 履歴 書類		
カレンダ: 標	準カレンダ - 日本	• 🗉 ک	
解決時間			
期限のタイプ:時	間・「時間		
			EBU" 7 (0)

## 「技術検証」アクティビティのアラームを定義する

[制限時間]タブページで指定した制限時間までに決定が下されなかった場合 に、催促のメッセージを送信するためのアラームを、アクティビティの詳細画 面の[**アラーム**]タブページで定義します。

№ ワークフローチャート 「購入依料	の承認'のアクティビティ		
名前: 技術検証	テンプレート:	• = •	新規作成( <u>N</u> )
タイプ: 質問	▼ 参照番号: REQAPPR-ACTIV0:	2	複製(∐)
全般 バラメータ 💩 時	間 🚳 アラーム 履歴 書類		
- アラーム 名前: 参照番号: - トリガするアクション	検証の依頼 REQAPPR-FUNCALRM01 タスク終了前の時間 0時間	•	
起動:	アクション	•	
アクション:	技術検証依頼	• 🗉 •	
検証の依頼			開じる( <u>C</u> )

処理を簡単にするために、このアラームでは**「技術検証依頼」**アクションをト リガすることにします。

ポップアップメニューの[**リンクレコードの追加**]コマンドを使って、さらに アラームを定義することもできます。

#### イベント

アクティビティを作成すると、質問に対する2種類の回答に対応する2つのシステムイベント(「**承認」**と「**却下」**)が作成されます。

#### <u> 承認</u> 却下

これらのイベントが発生すると、AssetCenterエージェントは、これらを[ワー クフローの基本イベント](SQL名:amWfOccurEvent)のテーブルに記録しま す。

AssetCenterまたはAssetCenter Serverにより、ワークフローのイベントの設定に応じて次のアクティビティがトリガされます。

- [処理](SQL名: seProcessingMode)フィールドが[イベントを保存して すぐに処理]または[イベントを保存しないですぐに処理]の場合、 AssetCenterが次のアクティビティをトリガします。
- [処理] (SQL名: seProcessingMode) フィールドが [イベントを保存して からサーバで処理]の場合、AssetCenter Serverが次のアクティビティをトリ ガします。

イベントの詳細画面の[全般]タブの[処理](SQL名: seProcessingMode) フィールドのデフォルト値は、[イベントを保存してからサーバで処理]で す。

#### 「財務検証」アクティビティを作成する

このアクティビティは、前のアクティビティと類似しています。

ワークフローの担当者は異なります。この場合、会社の会計監査役である木村 五郎氏です。この担当者は、前の手順で[ワークフローの役割]テーブル内に 作成されました。この担当者に通知するために、「財務検証依頼」アクション を選択します。

#### 「依頼の承認」アクティビティを作成する

依頼が2つの検証プロセスを問題なく通過すると、この依頼は承認されたことになります。

ワークフローチャートは、「依頼の承認」アクティビティで終了できます。

このアクティビティでは、依頼の詳細を変更して、依頼が承認されたことを指 定する必要があります。

このアクティビティは[自動アクション]タイプのアクティビティです。実行 されるアクションは、前の手順で作成した「購入依頼の承認」アクションです。

電 ワークフローチャート '購入依頼の承認' のアクティビティ	-OX
名前: 依頼の承認 テンプレート:	
タイプ: 自動アクション - 参照番号: REQ.APPR-A04	
全般 バラメータ 💩 時間 📸 アラーム 履歴 書類	
▼ アクションを即実行	
アクション(複数):	
+	
▼	
· · · · · · · · · · · · · · · · · · ·	更( <u>M</u> )

#### 「依頼の却下」アクティビティを作成する

「依頼の却下」アクティビティは「依頼の承認」アクティビティと類似しています。

この場合は、依頼の詳細を変更して、依頼が却下されたことを指定する必要が あります。実行されるアクションは、前の手順で作成した「購入依頼の却下」 アクションです。



## アクティビティと同時に作成されるイベントのパラメー タ

前段階でアクティビティが作成されると、以下のイベントも同時に作成されま す。

- 「技術検証」アクティビティ
  - 「承認」イベント
  - 「却下」イベント
- 「財務検証」アクティビティ
  - 「承認」イベント
  - 「却下」イベント
- 「依頼の承認」アクティビティ
  - 「実行済」イベント
- 「依頼の却下」アクティビティ
  - 「実行済」イベント

AssetCenter Serverに依存しないようにするために、各イベントを選択し以下の操作を実行します。

- 1 イベントの名前を右クリックします。
- 2 [イベントの詳細]メニューを選択します。

- 3 [全般]タブページを選択します。
- 4 **[処理]**(SQL名: seProcessingMode)フィールドで**[イベントを保存して すぐに処理]**を選択します。
- 5 [変更]をクリックします。

#### 開始イベントを作成する

ワークフローをトリガするイベントは、「**開始」**アクティビティに関連付けられます。

「開始」イベントを定義するには、「開始」アクティビティのイベント領域を マウスの右ボタンでクリックし、[イベントの詳細]を選択します。

1 この例では、ワークフローは、依頼の詳細画面の**[依頼ステータス]**(SQL 名: seStatus)フィールドが**[承認待ち]**になるとトリガされます。

従って、開始イベントは「**データベース」**タイプのイベントで、次のように [全般]タブページにパラメータを指定します。

◎ ワークフローチャート購入	依頼の承認'のイベント			
名前: 新規		タイプ: データベース	•	新規作成( <u>N</u> )
参照番号: REQ	_TO_VALIDATE			複製(∐)
全般パラメータ履り	歴 書類			
コンテキスト: am Re	equest			
処理: 귀~]	ントを保存してすぐに処理		•	
┌起動条件				
▼ ワークフローを再初期	HL.			
AQL条件:				
seStatus = 13			<u>^ </u>	
			<b>_</b>	
				閉じる( <u>C</u> )

- イベントの[全般]タブページにある[処理](SQL名: seProcessingMode)
   フィールドを[イベントを保存してすぐに処理]に設定します。
- 3 イベントの[パラメータ]タブページで、次の操作を行います。
  - 1 [更新] (SQL名: bUpdate) チェックボックスをオンにします。
  - 2 **[モニタするフィールド]**(SQL名: MonitFields)フィールドに [seStatus]フィールドを指定します。

138 > AssetCenter 4.2.0 - AssetCenterの高度な使い方

#### トランジションを作成する

アクティビティを作成したら、トランジションでリンクする必要があります。 トランジションを作成するには、次の手順に従います。

- 1 トランジションの開始イベントをクリックします。
- 2 マウスボタンを押したまま、ターゲットアクティビティまでイベントをド ラッグします。

この例では、次のトランジションを作成します。

- 開始イベントから「技術検証」アクティビティまで
- 「技術検証」アクティビティの「承認」イベントから「財務検証」アクティ ビティまで
- 「財務検証」アクティビティの「承認」イベントから「依頼の承認」アク ティビティまで
- 「技術検証」アクティビティおよび「財務検証」アクティビティの「却下」
   イベントから「依頼の却下」アクティビティまで

この結果、ワークフローチャートは次のようになります。

図 8.6. ワークフローチャート - 依頼の承認



#### ワークフローの起動例

ワークフローチャートが正しく機能することを確認する必要があります。 確認するには、次の操作を実行します。

- 1 [部署と従業員]テーブルに入力する
- 2 承認を受ける購入依頼を作成する
- 3 ワークフローを制御する

### [部署と従業員]テーブルに入力する

承認を受ける購入依頼を作成する前に、**[部署と従業員]**テーブルに依頼者と 依頼者の部署の責任者を定義する必要があります。責任者には、業務を遂行す るための適切な権限が必要です。

- 1 「情報システム部」に所属する依頼者「田中伸一」を作成します。
- 2 依頼者が購入依頼を作成できるように、田中伸一にログイン「田中」、パス ワード、およびユーザプロファイルを指定します(対応するレコードの詳細 画面の[プロファイル]タブページで指定)。デモ用データベース内のプロ ファイル「依頼者」を使用します。
- 3 **[電子メール]** (SQL名: EMail) フィールドに「AM:Tanaka」を入力しま す。
- 4 「**情報システム部」**の責任者は「鈴木俊彦」です。
- 5 後の操作を簡単にするために、「鈴木俊彦」にデータベースの管理者権限を 与えます。対応するレコードの詳細画面の[プロファイル]タプページを表 示し、[管理者権限](SQL名:bAdminRight)チェックボックスをオンに します。鈴木俊彦の[ログイン](SQL名:UserLogin)とパスワードを指定 します。
- 6 **[電子メール]** (SQL名: EMail) フィールドに「AM:Suzuki」を入力しま す。
- 7 従業員「山田啓介」を選択します。
- 8 **[電子メール]** (SQL名: EMail) フィールドに「AM:Yamada」を、**[ログ** イン] (SQL名: UserLogin) フィールドに「山田」を入力します。

#### 承認を受ける購入依頼を作成する

次の手順に従って、承認を受ける購入依頼を作成します。

- 1 「田中伸一」のログイン名を使ってデモ用データベースに接続します。
- 2 AssetCenterの[調達/購入依頼]メニューを選択します。
- 3 [新規作成]をクリックします。
- 4 [カスタム依頼を新規作成する]オプションを選択します。
- 5 [OK]をクリックします。
- 6 依頼の詳細画面の[**全般**]タブページにある[**依頼者**](SQL名: Requester) フィールドで、「田中」のレコードを選択します。
- 7 依頼の詳細画面の[依頼ステータス](SQL名: seStatus)フィールドを[承 認待ち]に設定します。
- 8 購入依頼の作成を確定します。開始イベントが発生し、[ワークフローの基本イベント](SQL名: amWfOccurEvent)テーブルにイベントが記録されます。

#### ワークフローを制御する

ワークフローが正常に機能することを確認するために、AssetCenterを起動し、 「田中伸一」の部署の責任者である「鈴木俊彦」のログインを使ってデモ用デー タベースに接続します。

#### ワークフローを表示する

作成した購入依頼の詳細画面を表示します。[ワークフロー]タブページには、 依頼の現在のワークフローが一覧表示されます。各ワークフローの詳細は、個々 のサブタブページに表示されます。

- サブタブページの左側には発生したイベントが一覧表示されます。
- 右側には、ワークフローのステータスが表示されます。ワークフローチャートの「技術検証」が点滅しているはずです。



#### 実行するタスクを表示する

1 [ツール/進行中のタスク]メニューを選択します。部署の責任者である 「鈴木俊彦」は、この方法で割り当てられたタスクの詳細を表示することが できます。



[全般]タブページには、「技術検証」アクティビティの[質問]タブページで定義した質問が表示されます。

[割当]タブページには、タスクの担当者と締切りが表示されます。この日付は、「技術検証」アクティビティの詳細画面の[制限時間]タブページで 指定した情報と、タスクの作成日(つまりトランジションの起動日)を使っ て計算されます。

[詳細]をクリックすると、タスクの発生元である依頼の詳細画面にアクセスできます。

2 [承認]ボタンまたは[却下]ボタンをクリックして、タスクを実行します。[全般]タブページに、決定に関する情報を入力することもできます。

#### ワークフローのコンテキスト

各ワークフローには固有のコンテキストがあります。

#### ワークフローのコンテキストを定義する

ワークフローチャートを定義する場合は、次の要素を定義します。

- ワークフローチャートの詳細画面上部にあるデフォルトのコンテキスト
- 全てのイベント、トランジション、およびワークフローのアクティビティの コンテキスト(トランジションの詳細画面、またはイベントおよびアクティ ビティの詳細画面の[全般]タブページで指定)。このコンテキストは、 ワークフローチャートのデフォルトのコンテキストにリンクされます。

いずれの場合も、コンテキストはテーブルです。

#### ワークフローが参照するオブジェクト

レコードが、ワークフローチャートで定義された起動条件を満たしたときに、 ワークフローがトリガされます。レコードは、開始イベントが参照するオブジェ クトです。

ワークフローの実行中に参照されるオブジェクトは、アクティビティ、イベント、およびワークフローのトランジションで定義したコンテキストによって変わります。

例:購入依頼が承認されると、ワークフローがトリガされます。この依頼に従っ て発注が作成されます。依頼R1が承認されると、このR1が開始イベントの参照 オブジェクトになります。その後参照オブジェクトは発注PO1、つまり購入依頼 から作成された発注になります。

## 特定のオブジェクトに対して進行中のワークフローを 制限する

# [1つの物件に付き1つのワークフローを起動]オプション(SQL名:bUniqueActive)

AssetCenterでは、ワークフローチャートの詳細画面の[全般]タブページにある[1つの物件に付き1つのワークフローを起動]オプションを使って、選択しているオブジェクトで同時に実行できるワークフロー数を制限できます。 あるオブジェクトで2番目のワークフローを起動する「開始」アクティビティの出力イベントが発生した場合、[1つの物件に付き1つのワークフローを起動] オプションと[進行中のワークフローがある場合にワークフローを再度初期化] オプション(SQL名:bReinitialize)(イベントの詳細画面の[全般]タブページ内)によって結果が決まります。

オプションの選択に応じた結果は、次の表の通りです。

		ワークフローチャートの [ 全般 ] タブペー ジの [ 1つの物件に付き1つのワークフロー を起動 ] オプション
		オンオフ
「開始」アクティビ	オン	そのオブジェクトに対して既に実行中の
ティの出力イベント		ワークフローがある場合、そのワークフ
の[全般]タブペー		ローは停止し、新しいワークフローが開始
ジにある [進行中の		します。
ワークフローがある	オフ	そのオブジェクトに 新しいワークフロー
場合にワークフロー		対して既に実行中のが作成されます。
を再度初期化 ] オプ		ワークフローがある
ション		場合、イベントは無
		視されます(新しい
		ワークフローは開始
		しません)。

#### 適用例

購入依頼の承認処理を目的とするワークフローチャートの場合は、次のように 設定すると便利です。

[1つの物件に付き1つのワークフローを起動]オプションをオンにして、対象となる購入依頼を単一の承認プロセスで処理する。

 開始イベントで[進行中のワークフローがある場合にワークフローを再度初 期化]オプションをチェックして、購入依頼の構成が変更された場合にワー クフローを再起動する。

## ワークフローの役割

特定のワークフローのアクティビティから発生するタスクは、担当者が実行す る必要があります。

💋 注意:

アクティビティの担当者は、[質問]または[ユーザのアクション]タイプの アクティビティだけに必要です。[自動アクション]または[テスト/スクリ プト]タイプのアクティビティには不要です。

アクティビティの担当者は、**[ワークフローの役割]**(SQL名:amWfOrgRole) テーブル(SQL名:amWfOrgRole)で選択します。**[ワークフローの役割]**テー ブルにアクセスするには、**[ツール/ワークフロー/役割]**メニューを選択し ます。

#### ワークフローの役割のタイプ

ワークフローの役割には次のタイプがあります(ワークフローの役割の詳細画面 / **[タイプ]**(SQL名:seType)フィールドで指定)。

- [個人に割当]
- [特定の個人に割当]
- [グループに割当]

#### [個人に割当]

この場合担当者は、[部署と従業員]テーブルから直接選択されます。
例

◎ 役割 '会計監査役' の詳細					_ 🗆 🗡
△役割		△コンテキ	スト	+	新規作成(N)
責任者の承認		amReques	t 👘		
財務の承認		amReques	t I		複製(山)
依頼者の部署の責任者		am Reques	t	_	第11日本(D)
会計監査役			_	<u> </u>	FUDA (D)
4				Ŧ	
		_ ⊗_	6/6		
····································	参照番号:	D001002		-	
コンテキスト: 📰 (テーブルなし) 🔹	タイプ:	個人に割当		•	
担当者:木村五郎			۹ 🗉	•	
					閉じる(0)

# [特定の個人に割当]

この場合担当者は、**[部署と従業員]**テーブルに属しますが、スクリプトによって決定されます。

例

🍋 役割 '依頼者の部署の責任者' の詳細					_ 🗆 ×
△役割		△コンテキ:	スト	+	新規作成(N)
依頼内容の承認		am Request			
責任者の承認		amRequest			複製(U)
財務の承認		am Request		_	#URA(D)
依頼者の部署の責任者		am Request		-	1)禄(世)
				+	
		_ ⊗	5/5		
役割: 依頼者の部署の責任者	参照番号:	D001001			
コンテキスト: 📰 依頼 (am Request) 🔹	タイブ:	特定の個人に割当		•	
スクリプト:					
Retval=[Requester.Parent.Supervisor]			<b>A</b>	٩	
			<b>_</b>		
र			Þ		<u>閉じる(C)</u>

# [グループに割当]

この場合担当者は、従業員グループ(SQL名:amEmplGroup)から選択されます。

## アクティビティの担当者を定義する

[質問]タブ([質問]タイプのアクティビティ)または[アクション]タブ ([ユーザのアクション]タイプのアクティビティ)の[担当者](SQL名: Assignee)フィールドを使って、[質問]タイプまたは[ユーザのアクション] タイプのアクティビティの担当者を定義できます。

# ワークフローのアクティビティ

アクティビティは、次の2つのグループに大別できます。

- ユーザの操作を必要とするアクティビティ: [質問]および[ユーザのアク ション]タイプのアクティビティ(アクティビティの詳細画面の上部にある [タイプ](SQL名:seType)フィールド)
- 自動的に実行されるアクティビティ:[自動アクション]および[テスト/ス クリプト]タイプのアクティビティ

アクティビティの[**タイプ**]フィールドの値によって、アクティビティの詳細 画面に表示されるタブが決まります。

ここでは、次の内容について説明します。

- 「質問」タイプのアクティビティ
- 「ユーザのアクション」タイプのアクティビティ
- 「自動アクション」タイプのアクティビティ
- 「テスト / スクリプト」タイプのアクティビティ
- [開始]アクティビティ
- アクティビティテンプレート
- アクティビティをトリガする

## 「質問」タイプのアクティビティ

「質問」タイプのアクティビティには、**[担当者]**(SQL名:Assignee)フィールドで指定したユーザの操作が必要になります。

- [質問]タイプのアクティビティでは、次の内容を定義します。
- ・
   ・
   質問または指示
- 答の選択肢

例

- 購入承認プロセスで、従業員からの依頼が部署の責任者に発信されます。
- [質問]タイプのアクティビティを確認プロセスとして使い、タスクが実行 されたことを確認することもできます。この場合、答の選択肢は1つだけで す。

**[タイプ]**(SQL名:seType)フィールドを**[質問]**に設定します。**[質問]**タ ブページが表示されます。

次の内容を指定します。

 ワークフローの役割テーブルに含まれている担当者に対応するレコード。この担当者には、AssetCenterで定義するアクションを介して通知できます。通知することを指定するには、[質問]タブページの[通知する](SQL名: bNotifAssignee)フィールドで「はい」を選択します。

#### 💋 注意:

担当者に通知するアクションは、実行するタスクが作成された時点、つまり アクティビティを発生させるトランジションが起動された時点ですぐにトリ ガされます。

担当者は、[ツール/進行中のタスク]メニューを使って、実行するタスクの詳細情報にアクセスします。

- 2 質問または指示の本文
- 3 答の選択肢。答の説明は個々のサブタブページに入力します。答は、説明とそのSQL名で識別されます。答を追加、複製、または削除するには、サブタブのラベル部分をマウスの右ボタンでクリックし、ポップアップメニューから[リンクレコードの追加]、[リンクレコードの複製]、または[リンクの削除]を選択します。

💋 注意:

答が返ってくると、自動的にアクティビティの出力イベントが作成されます。

## 「ユーザのアクション」タイプのアクティビティ

「質問」タイプのアクティビティには、**[担当者]**(SQL名:Assignee)フィールドで指定したユーザの操作が必要になります。

次の内容を定義します。

- 指示の内容
- 実行するウィザード

[**タイプ**]フィールドを[**ユーザのアクション**]に設定して、[**アクション**] タブを表示します。

次の内容を指定します。

- 指示の内容
- 実行するウィザード
- [ワークフローの役割]テーブルに含まれている担当者に対応するレコード。この担当者には、AssetCenterのアクションを介して通知できます。通知

することを指定するには、**[アクション]**タブページの**[通知する]**(SQL 名:bNotifAssignee)フィールドで「はい」を選択します。

#### 💋 注意:

担当者に通知するアクションは、実行するタスクが作成された時点、つまり アクティビティを発生させるトランジションが起動された時点ですぐにトリ ガされます。

担当者は、[ツール/進行中のタスク]メニューを使って、実行するタスクの詳細情報にアクセスします。

💋 注意:

「Executed」(実行済)イベントが、アクティビティの出力イベントとして自動的に作成されます。

例:受領を管理する時に、納品待ちの発注明細をすべて受領するのか、部分的 に受領するのかをウィザードを使って管理できます。

# 「自動アクション」タイプのアクティビティ

「自動アクション」タイプのアクティビティは、AssetCenterまたはAssetCenter Serverによって自動的に実行されます。

#### 説明

[自動アクション]タイプのアクティビティには、実行されるアクションが一 覧表示されます。

例:「資産の移動」処理では、[自動アクション]タイプのアクティビティにより、移動した親資産に属するすべての資産の設置場所が自動的に変更されます。

[タイプ]フィールドを[自動アクション]に設定して、[アクション]タブ ページを表示します。

実行するアクションのリストが表示されます。

💋 注意:

「Executed」(実行済)イベントが、アクティビティの出力イベントとして自動的に作成されます。

## 実行

アクティビティを発生させるトランジションを起動するワークフローエンジン は、アクティビティのアクションを即時に実行します。トランジションをトリ ガするイベント用に選択した処理モードに従って、AssetCenter Serverまたは AssetCenterのエージェントがアクションを実行します。

- [アクションを即実行](SQL名: bExecImmediately)オプションをオンにした場合は、アクティビティを発生させるトランジションを起動するワークフローエンジンにより、アクティビティのアクションが自動的に実行されます。
- それ以外の場合は、タスクは次の検証サイクル中にAssetCenter Serverによっ て実行されます。

# 「テスト / スクリプト」タイプのアクティ**ビ**ティ

「自動アクション」タイプのアクティビティは、AssetCenterまたはAssetCenter Serverによって自動的に実行されます。

#### 説明

このタイプのアクティビティには、スクリプトとその実行結果の選択肢を定義します。

例: 在庫や購入依頼管理を行うときに、テスト / スクリプトタイプのアクティ ビティを使って、発注明細が参照している品目が在庫にあり、予約されていな いかどうかを確認できます。予約されていない場合は、依頼者に在庫の品目を 予約するかどうかを尋ねる[質問]タイプのアクティビティをトリガさせるこ とができます。

**[タイプ]**を**[テスト/スクリプト]**に設定して、**[テスト]**タブを表示します。 次の内容を指定します。

- 実行するテストスクリプト
- テスト結果の選択肢。各結果の説明は個々のサブタブページに入力します。
   結果は説明とSQL名で識別されます。結果の選択肢を追加、複製、または削除するには、サブタブのラベル部分をマウスの右ボタンでクリックし、ポップアップメニューから[リンクレコードの追加]、[リンクレコードの複製]、または[リンクの削除]を選択します。

#### 💋 注意:

注意:各結果のSQL名は、テストスクリプトの戻り値と対応させる必要があり ます。

💋 注意:

結果が出るたびに、自動的にアクティビティの出力イベントが作成されます。

## 実行

アクティビティを発生させるトランジションを起動するワークフローエンジン は、アクティビティのアクションを即時に実行します。トランジションをトリ ガするイベント用に選択した処理モードに従って、AssetCenter Serverまたは AssetCenterのエージェントがアクションを実行します。

- [アクションを即実行](SQL名: bExecImmediately)オプションをオンに した場合は、アクティビティを発生させるトランジションを起動するワーク フローエンジンによって、アクティビティのアクションが自動的に実行され ます。トランジションを起動するイベント用に選択した処理モードに従っ て、AssetCenter ServerまたはAssetCenterのエージェントがアクションを実行 します。
- それ以外の場合は、タスクは次の検証サイクル中にAssetCenter Serverによっ て実行されます。

# [開始]アクティビティ

[開始]アクティビティは、ワークフローチャートの開始点です。 開始アクティビティは必須で、ワークフローチャートの作成時に自動的に作成 されます。詳細を編集することはできません。 開始アクティビティは実行する作業を定義しません。 [開始]アクティビティの出力イベントによってワークフローがトリガされま す。

# アクティビティテンプレート

アクティビティテンプレートを使うと、ワークフローチャートのアクティビティ を迅速に作成できます。

アクティビティテンプレートは、**[アクティビティ]**(SQL名 : amWfActivity) テーブルに保存されます。

アクティビティテンプレートのリストを開くには、**[ツール/ワークフロー/ アクティビティテンプレート]**メニューを選択します。

### 💋 注意:

注意:アクティビティテンプレートの詳細画面に含まれている情報(アクティ ビティのタイプなど)を、このテンプレート(アクティビティの詳細画面の[**テ** ンプレート](SQL名:Template)フィールドで指定)を参照しているアクティ ビティに自動的にコピーするには、AssetCenter管理者がアクティビティの詳細 画面のフィールドとリンクに適切なデフォルト値を定義する必要があります。

## アクティビティをトリガする

アクティビティを起動するには、アクティビティの詳細画面の[**全般**]タブページにある[**入力条件**](SQL名:selnCond)フィールドに入力する必要があります。これは、アクティビティを起動するトランジションに関する条件です。

- アクティビティを起動するトランジションが1つだけの場合は、(AssetCenter またはAssetCenter Serverによって)トランジションを起動するだけで、アク ティビティがトリガされます。
- アクティビティを起動するトランジションが複数ある場合は、次のように処理されます
  - アクティビティの入力条件が[AND]の場合、アクティビティをトリガ するには、すべてのトランジションが起動されなければなりません。
  - アクティビティの入力条件が[OR]の場合、トランジションを1つ起動 するだけで、アクティビティがトリガされます。

#### 💋 注意:

アクティビティの入力条件が複雑な場合([AND]と[OR]の組合せ)は、連続した中間[**テスト/スクリプト]**タイプのアクティビティを作成して代用する こともできます。

# タスク

ここでは、ワークフローのタスクを作成し、実行する方法について説明します。

### タスクを作成する

タスクを発生させるトランジションが起動すると、トランジションを起動した ワークフローエンジンによって、実行するタスクが自動的に作成されます。

アクティビティの詳細画面の**[全般]**タブページにある**[タスクを保存]**(SQL 名:bLogWorkItem)フィールドで選択したオプションに従って、タスクは**[ワー クフローのタスク]**(SQL名:WkEle)テーブルに記録されます。

[タスクを保存]オプションは、次の場合に自動的にオンになります。 • [質問]または[ユーザのアクション]タイプのアクティビティの場合

[アクションを即実行](SQL名: bExecImmediately)オプションが選択されていない[自動アクション]、または[テスト/スクリプト]タイプのアクティビティの場合

#### 🌻 警告:

タスクが記録されないと、このタスクに関連付けられているワークフローのア ラームを作成することはできません。[タスクを保存]オプションを選択しな いと、アクティビティの詳細画面に[制限時間]および[アラーム]タブが表 示されません。

タスクは、ユーザの操作が必要かどうかによって、異なる方法で実行されます。

# 自動アクションまたはテスト / スクリプトタイプのア クティビティ

[アクションを即実行](SQL名:bExecImmediately)オプションが選択されている[自動アクション]、または[テスト/スクリプト]タイプのアクティビティから発生したタスクは、タスクを生成するトランジションの起動元のワークフローエンジンによって実行されます。このエンジンは、AssetCenter ServerまたはAssetCenterエージェントです。

それ以外の場合は、AssetCenter Serverによって、ワークフローのタスクを実行す る必要があるかどうかが定期的に検証され、必要に応じて実行されます。

AssetCenter Serverがワークフロー関数をモニタする頻度は、AssetCenter Serverの オプションで定義します。

## 進行中のタスクのリストを表示する

実行する必要のあるタスクのリストを表示するには、[ツール/進行中のタス ク]メニューを選択します。

リストには、データベースに接続しているユーザに応じて、次のタスクが表示 されます。

- AssetCenter管理者は、すべてのワークフローの進行中の全タスクを表示できます。
- ワークフローの担当者は、次のタスクを表示できます。

- 自分が実行しなければならないタスク
- 自分には割り当てられていないけれども、自分が属するグループに割り 当てられているタスク

管理者は、ワークフローチャートの詳細からも特定のアクティビティに関する 進行中のタスクのリストにアクセスできます。次の手順に従います。

フローチャート内のアクティビティをマウスの右ボタンでクリックします。
 ポップアップメニューから「タスク」を選択します。

#### 💋 注意:

表示されるリストは、**[ワークフローのタスク]**(SQL名:amWfWorkItem) テーブルだけに限定されたビューです。実行しなければならないタスクが表示 されます。

### ユーザタスクを実行する

ワークフローの担当者は、**[ツール/進行中のタスク]**メニューを選択して、 実行するタスクのリストにアクセスできます。

#### 🧶 警告:

データベースに接続しているユーザがAssetCenter管理者の場合は、[ツール/ 進行中のタスク]メニューを選択すると、実行する必要のあるすべてのタスク が表示されます。接続ユーザが管理者ではない場合は、[ツール/進行中のタ スク]メニュー項目を選択すると、ユーザに割り当てられたタスクまたはユー ザが属するグループに割り当てられたタスクのみが表示されます。

タスクが参照するオブジェクトの詳細にアクセスするには、[詳細]をクリックします。

指定されたタスクを実行するには、タスクの[全般]タブページを表示します。

- タスクの結果発生するアクティビティが[質問]タイプのアクティビティの場合、[全般]タプには質問または手順のテキストが表示されます。答に対応するボタンがあるので、適切なボタンをクリックします。必要な場合は、決定に関するコメントも入力できます。
- [ユーザのアクション]タイプのアクティビティの場合は、[ウィザード] ボタンをクリックして、実行するウィザードを起動します。

## ユーザにタスクを割り当てる

ユーザタスクの割当てに関する情報は、タスクの詳細画面の**[割当]**タブに表示されます。

必要な権限がある場合は、ユーザタスクの割当てを変更できます。

- [割当] (SQL名: seAssignment) フィールドの値
- タスクの担当者

## ワークフローのタスクを管理する

ワークフロータスクの管理に関する情報は、タスクの詳細画面の[管理]タブ ページに表示されます。

情報を表示できるのは、管理者権限を持つユーザのみです。

## イベント

イベントはアクティビティに関連付けられます。イベントは、他のアクティビ ティへのトランジションを起動します。

アクティビティのレベルには、3種類のシステムタイプイベントがあります。イ ベントのシステムタイプは、イベントの詳細画面の**[システムタイプ]**(SQL 名:seType)フィールドで定義します。

- [**システム**]イベント
- [ユーザ]イベント
- [**アラーム**]イベント

# システムイベント

[システム]イベントは、アクティビティの作成および変更時にAssetCenterによって自動的に定義されます。

これらは、アクティビティで実行するタスクの各結果に対応します。

- [質問]タイプのアクティビティに対する答
- [**テスト/スクリプト**]タイプのアクティビティの結果
- [ユーザのアクション]または[自動アクション]タイプのアクティビティの場合は、「Excuted」(実行済)イベント

例:アクティビティで、答が「はい」または「いいえ」となる質問を尋ねる場合、アクティビティのレベルでは、「はい」と「いいえ」の2つのシステムイベントが作成されます。

## アラームイベント

アクティビティ用アラームのイベントは、イベントを起動するアクティビティ 用アラームを定義すると作成されます。

このようなアラームは、アクティビティの詳細画面の**[アラーム]**タブで定義します。イベントの名前はアラームと同じになります。

## ユーザイベント

ユーザイベントは、アクティビティ内で実行されるタスクとは関係ありません。 これらは、グラフィカルワークフローエディタから手動で作成します(**[イベ** ントの追加]ポップアップメニューを使います)。

💋 注意:

[開始]アクティビティに関連付けられるイベントはユーザイベントです。

次の2つのタイプの**[ユーザ]**イベントがあります(イベントの詳細画面上部の **[タイプ]**(SQL名:seMonitoringType)フィールドで指定します)。

- ・ [データベース]
- [定期的]

## [データベース]タイプのイベント

[**データベース**]タイプのイベントでは、特定のレコードでワークフローを起動できます。

[データペース]タイプのイベントは、次の場合に発生します。

- [全般] タブページで指定した一般的な起動条件が満たされた場合
- モニタしているレコードで特定のトリガパラメータが確認された場合

[**データベース**]タイプのイベントをトリガするパラメータは、イベントの詳細画面の[**パラメータ**]タブページに入力します。次の情報を指定します。

- モニタするレコード(これらのレコードは、コンテキストレコードまたはリンクしているレコードのテーブル内のレコードでも構いません)。モニタするレコードが、コンテキストとして指定されているテーブルにリンクするレコードの場合は、対応するリンクを[リンク/コンテキスト](SQL名: LinkToMonitTable)フィールドで指定します。
- モニタ中のレコードに関するイベントの起動条件。起動の条件を指定するには、次の方法があります。
  - [挿入](SQL名:blnsert)、[更新](SQL名:bUpdate)、および[削
     除](SQL名:bDelet)チェックボックスを必要に応じてオンにします。

AssetCenter

[挿入]チェックボックスをオンにすると、作成されたレコードをモニ タします。

[更新] チェックボックスをオンにすると、変更されたフィールドをモ ニタします。モニタするフィールドは、[モニタしたフィールド](SQL 名:MonitFieldsbox)フィールドで指定できます。複数のフィールド名を 指定する場合は、カンマを使って区切ります。フィールドを空のままに すると、変更されたフィールドはモニタされません。

[**削除**] チェックボックスをオンにすると、破棄されたレコードがモニ タされます。

#### ! 警告:

コンテキストが参照するオブジェクトの破棄をイベントの起動条件にす ることはできません。

 [スクリプト](SQL名:memScript)ゾーンにスクリプトを入力します。 スクリプトを入力し、[挿入]、[更新]、[削除]のチェックボック スを必要なだけオンにすると、スクリプトによって起動条件が制限され ます。

例:既存の依頼の総額が変更されたときにイベントをトリガする場合は、[パ ラメータ]タブに次のように入力します。

#### 図8.7. [データベース]タイプアクションの[パラメータ]タブページ

全般パラメータ展歴書	類
-+=	
モニタするテーブル:	amTicketHistLine
 ┌モニタするレコードの起動条件	
☑ 挿入     ☑ 更新	前除
モニタするフィールド:	dtStart, tsProcessTime, lAuthorId, lTicketId 🔍 🤦
イベントスクリプト:	
x	R N N

[データベース]タイプのイベントが発生すると、そのイベントは、発生した AssetCenterクライアントマシンによって処理されます。処理方法は、イベント の詳細画面の[全般]タブページにある[処理](SQL名: seProcessingMode) フィールドで選択したオプションによって決まります。 詳しくは、本マニュアルの「**ワークフロー」**の章、「イベントの処理」の節を 参照してください。

## [定期的]タイプのイベント

[定期的]タイプのイベントは、テーブルで選択された複数のレコードを対象 とします。このタイプのイベントを使うと、選択した各レコードに対して定期 的にワークフローをトリガできます。

例:毎月、属性が「PC」の資産の残存価額を更新します。

[定期的]タイプのイベントは、[全般]タブページに指定されている起動条件が満たされたときに発生します。

この場合、AssetCenter Serverによってイベントがトリガされます。

AssetCenter Serverが[定期的]タイプのイベントをトリガする頻度は、イベントの詳細画面の[パラメータ]タブページで定義します。

イベントの処理方法については、本マニュアルの「ワークフロー」の章、「イ ベントの処理」の節を参照してください。

### 起動の一般的な条件

すべてのタイプのイベントの起動条件は、[全般]タブページで定義できます。

### [AQL条件] (SQL名: AQLCond)

[AQL条件]フィールドでは、ワークフローチャートに使われるレコードを指 定します。

## [進行中のワークフローがある場合にワークフローを再 度初期化](SQL名:bReinitialize)

#### 💋 注意:

[進行中のワークフローがある場合にワークフローを再度初期化]オプションは、「開始」アクティビティの結果起動されるイベントの詳細画面にのみ表示されます。

[進行中のワークフローがある場合にワークフローを再度初期化]オプションは、[開始]アクティビティの出力イベントが、同じワークフローチャートの別のワークフローで既に使われているデータベースオブジェクトを必要とする場合の処理を決定します。

このオプションだけでなく、ワークフローチャートの[全般]タブページにある[1つの物件に付き1つのワークフローを起動](SQL名:bUniqueActive)オプションも、処理方法を決定します。

オプションの選択に応じた結果は、次の表の通りです。

#### 表 8.1. ワークフローを制限する場合

		ワークフローチャー  ジの [ 1つの物件に付 を起動 ] オプション	トの [ 全般 ] タブペー き1つのワークフロー
		オン	オフ
「開始」アクティビ ティの出力イベント の[全般]タブペー ジにある[進行中の	オン	そのオブジェクトに対 ワークフローがある ローは停止し、新しい します。	対して既に実行中の 場合、そのワークフ ヽワークフローが開始
ワークフローがある 場合にワークフロー を再度初期化 ] オプ ション	オフ	そのオブジェクトに 対して既に実行中の ワークフローがある 場合、イベントは無 視されます(新しい ワークフローは開始 しません)。	新しいワークフロー が作成されます。

# イベントの処理

ー般的な起動条件が満たされた場合、イベントは、次の指定に従って処理され ます。

- イベントの「タイプ」(イベントの詳細画面上部の[タイプ](SQL名: seMonitoringType)フィールド)
- イベントの詳細画面の[全般]タブページにある[処理](SQL名: seProcessingMode)フィールドで選択されるオプション

イベントの様々な処理方法は、以下の表の通りです。

表 8.2. イベントの様々な処理方法

	[イベントを保存し てからサーバで処 理]	[イベントを保存し てすぐに処理]	[イベントを保存し ないですぐに処理]
[ 定期的 ] タイプの イベント	起動条に、 AssetCenter Serverに よってイベントがト リガガ気度細す。 ベントガト ガの頻詳一夕」タブペー ジで定するとすぐに、 AssetCenter Serverに より、 [amWfOccurEvent] というSQL名のティズ なままtCenter Serverに より、 にイベントがト シンジションは、 AssetCenter Serverに よす。 AssetCenter Serverに よす。 人SSETCENTER を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を を見ついた を見ついた を見ついた を見ついた を を見ついた を見ついた を見ついた を見ついた を見ついた を を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を見ついた を を を を を を を を を を を を を を を を を を を	起動条件が満たされ たとさに、 AssetCenter Serverに よってイベントがト リガの見は、イベン トリガの頃細面の[パ ラメータ]タブペー ジで定義します。 発生するとすぐに、 AssetCenter Serverに より、 [amWfOccurEvent] というSQL名のテー ブルにイベントが保 存されます。 トランジションは、 AssetCenter Serverに よって印座に起動されます。	起動条件が満たされ AssetCenter Serverに よってイベントがト リガの頻細面の[パ ラメータ]タブページ イベントがア ラメータ]タブページ イベントは AssetCenter Serverに よって、 [amWfOccurEvent] といらSQL名のテー ブルにしい。ただし、 トランジションは、 AssetCenter Serverに ようシンジションは、 AssetCenter Serverに ようシンジションは、 AssetCenter Serverに ようシンは、 AssetCenter Serverに ようシンジションは、 AssetCenter Serverに ようシジションは、 AssetCenter Serverに ようう。

	[イベントを保存し てからサーバで処 理]	[イベントを保存し てすぐに処理]	[イベントを保存し ないですぐに処理]
「データベース」タ イプのイベント、ま たはAssetCenterに よってトリガされる システムイベント ([質問]または [ユーザのアクショ ン]タイプのアク ティビティの結果、 AssetCenterによって 実行される[自動ア クション]または [テスト/スクリプ ト]タイプのアク ティビティの結果)	イベントが発生する とすぐに、 AssetCenterクライア ントマシンにより、 [amWfOccurEvent] というSQL名のテー ブルにイベントが保 存されます。 トランジションは、 AssetCenter Serverに よって後で起動され ます(AssetCenter Serverがトランジ ションの起動をモニ タする頻度は、 AssetCenter Serverの オプションで定義し ます)。	イベントが発生する とすぐに、 AssetCenterクライア ントマシンにより、 [amWfOccurEvent] というSQL名のテー ブルにイベントが保 存されます。 トランジションは AssetCenterクライア ントマシンによって 即座に起動されま す。	イベントが発生して も、 [amWfOccurEvent] というSQL名のテー ブルにイベントは保 存されませんが、ト ランジションは、 AssetCenterクライア ントマシンによって 即座に起動されま す。
AssetCenter Serverに よってトリガされる システムイベント (AssetCenter Server によって実行される [自動アクション] または[テスト/スク リプト]タイプのア クティビティの結 果)またはアクティ ビティアラームに関 するイベント	イベントが発生する とすぐに、 AssetCenter Serverに より、 [amWfOccurEvent] というSQL名のテー ブルにイベントが保 存されます。 トランジションは、 AssetCenter Serverに よって後で起動され ます (AssetCenter Serverがトランジ ションの起動をモニ タする頻度は、 AssetCenter Serverの オプションで定義し ます)。	イベントが発生する とすぐに、 AssetCenter Serverに より、 [amWfOccurEvent] というSQL名のテー ブルにイベントが保 存されます。 トランジションは、 AssetCenter Serverに よって即座に起動さ れます。	イベントが発生して も、イベントは AssetCenter Serverに よって、 [amWfOccurEvent] というSQL名のテー ブルに即座に記録さ れません。ただし、 トランジションは、 AssetCenter Serverに より即座に起動され ます。

表 8.3. イベントの様々な処理方法

	[イベントを保存してすぐ	[イベントを保存しないで
	「処理」	すぐに処理]
[定期的]タイプのイベン	起動条件が満たされたとき	起動条件が満たされたとき
F	に、AssetCenter Serverによっ	に、AssetCenter Serverによっ
	てイベントがトリガされま	てイベントがトリガされま
	す。トリガの頻度は、イベ	す。トリガの頻度は、イベ
	ントの詳細画面の <b>[パラ</b>	ントの詳細画面の <b>[パラ</b>
	<b>メータ</b> ] タブページで定義	<b>メータ ]</b> タブページで定義
	します。	します。
	発生するとすぐに、	イベントが発生しても、イ
	AssetCenter Serverにより、	ベントはAssetCenter Server
	[amWfOccurEvent]という	によって、
	SQL名のテーブルにイベン	[amWfOccurEvent]という
	トが保存されます。	SQL名のテーブルに即座に
	トランジションは、	記録されません。ただし、
	AssetCenter Serverによって	トランジションは、
	即座に起動されます。	AssetCenter Serverにより即
		座に起動されます。
[データベース]タイプの	イベントが発生するとすぐ	イベントが発生しても、
イベント、または	に、AssetCenterクライアン	[amWfOccurEvent]という
AssetCenterによってトリガ	トマシンにより、	SQL名のテーブルにイベン
されるシステムイベント	[amWfOccurEvent] という	トは保存されませんが、ト
([質問]または[ユーザ	SQL名のテーブルにイベン	ランジションは、
のアクション」タイフのア	トが保存されます。	AssetCenterクライアントマ
クティビティの結果、	トランジションは	シンによって即座に起動さ
AssetCenterによって実行さ	AssetCenterクライアントマ	れます。
れるし自動アクションしま	シンによって即座に起動さ	
	れます。	
タイノのアクティヒティの 対目)		
航末) AssatCantar Sanvarにトゥア	イベントが発生するとすぐ	イベントが発生してキーイ
トリガネカスシステムイベ	「「く」「小先王」。 こ AssetCenter Serverによ	イントけAssetCenter Server
<b>V b (</b> AssetCenter Server	1) [amWfOccurEvent]	によって
よって実行される「自動ア	いうSQL名のテーブルにイ	「amWfOccurEvent」という
クション1または「テスト/	ベントが保存されます。	SQL名のテーブルに即座に
スクリプト1タイプのアク		記録されません。ただし、
ティビティの結果)または	$1 \mathcal{I} \mathcal{I} \mathcal{I} \mathcal{I} \mathcal{I} \mathcal{I} \mathcal{I} \mathcal{I}$	トランジションは、
アクティビティアラームに	印成にお動なわます	AssetCenter Serverにより即
関するイベント		座に起動されます。

これらの様々なモードを使って、ワークフローの実行方法を正確に指定できます。

次のレベルで行う選択に従って、

- イベントタイプ
- イベント処理モード
- アクティビティ

同期および非同期ワークフローチャートを定義できます。また、両方のアプロー チを組み合せることもできます。

### 応用:同期ワークフローチャートを実行する

同期ワークフローチャートを作成するには、次の内容を定義する必要がありま す。

- [イベントを保存してすぐに処理](イベントの詳細画面の[全般]タブ ページ/[処理](SQL名: seProcessingMode)フィールド)を指定した [データベース]タイプのイベント
- そのイベントによってトリガされる、[アクションを即実行](SQL名: bExecImmediately)オプションを指定した[自動アクション]、または[テ スト/スクリプト]タイプのアクティビティ。

例

下記のワークフローチャートを使うと、資産の設置場所が変更されるのと同時 に、その従属資産が自動的に同じ場所へ移動するようになります。

#### 図 8.8. 同期ワークフローチャートの例



この場合、資産の設置場所を変更し[変更]をクリックすると、次の処理が行われます。

1 データベーストランザクションが開始します。

AssetCenterの高度な使い方

- 2 資産の設置場所が変更されます。
- 3 ワークフローが開始します。
- 4 ワークフローのトランジションが起動します。
- 5 従属資産の設置場所が変更されます。
- 6 トランザクション全体が検証されます。

1つの手順にエラーが発生すると、資産と従属資産とも設置場所は変更されません。

処理が正常に終了すると、すべての設置場所が変更されます。

一方、下に示すように、非同期ワークフローチャートを使って同じ処理を実行 すると、エラーが発生した場合は、従属資産の設置場所を変更せずに、資産の 設置場所を変更することができます。

#### 図 8.9. 非同期ワークフローチャートの例



## 終了イベント

### 定義

終了イベントが発生すると、実行するタスクが残っている場合もワークフロー は終了します。 例 AssetCenter

図 8.10. 終了イベント付きのワークフローチャート



上記のワークフローが発生し、次の状態になったとします。

- アクティビティ1の出力イベントが発生してアクティビティ2がトリガされ、 実行するタスクが作成された。
- アクティビティ3の終了イベントが発生した。

この場合、アクティビティ2のタスクが実行されていない場合も、ワークフロー は終了します。

### イベントを終了イベントとして指定する

ワークフローチャートの[**アクティビティ**]タブページにあるグラフィカルエ ディタを使って、ワークフローチャートを作成する場合は、次の方法で終了イ ベントを指定します。

1 ワークフローチャート内のイベントをマウスの右ボタンでクリックします。
 2 ポップアップメニューから[終了イベント]を選択します。

## ワークフローのトランジション

トランジションは、あるアクティビティから別のアクティビティへ出力イベン トをリンクします。

イベントは複数のトランジションにリンクできます。

必要に応じて、トランジションの詳細画面の[AQL条件](SQL名:AQLCond) フィールドで、トランジションの起動条件を指定できます。

## ワークフローのアラームと制限時間

ワークフローのアクティビティごとに、次の要素を定義できます。

- 実施の期限
- 実施の期限またはデータベースに格納されている日付にリンクするアラーム。このアラームによりアクションが起動します。

### ! 警告:

アクティビティの詳細画面の[全般]タブページにある[タスクを保存](SQL 名:bLogWorkItem)オプションをオンにしないと、制限時間とアラームは定義 できません。

### 制限時間

ワークフローのアクティビティ実施の制限時間は、アクティビティの詳細画面の[**制限時間**]タブページで定義します。

#### 💋 注意:

アクティビティの詳細画面の[制限時間]タブページは、同じ画面の[全般] タブページにある[タスクを保存](SQL名:bLogWorkItem)オプションをオ ンにしないと表示されません。

この制限時間は、アクティビティがトリガされる時間に基づいて定義されます。 制限時間は、業務用カレンダに関連付けられます。

期間を指定するか、または次の3つの定義済みオプションから選択します。

- [翌日]
- [今週末]
- [今月末]

! 警告:

期間を入力すると、AssetCenterは業務時間で指定していると認識し、時間単位 に変換します。例えば、「2日」と入力すると、48時間と解釈されます。

## ワークフローのアラーム

アクティビティの詳細画面の**[アラーム]**タブページで、各アクティビティに アラームを関連付けることができます。

#### 💋 注意:

アクティビティの詳細画面の**[アラーム]**タブページは、同じ画面の**[全般]** タブページにある**[タスクを保存]**(SQL名:bLogWorkItem)オプションをオ ンにしないと表示されません。

### 制限時間

アラームをトリガする制限時間は、次のいずれかの方式で定義できます。

- データベースに格納されている日付から経過した期間で指定([タスクを開始してから経過した時間]タイプ)
- データベースに格納されている日付前の期間で指定([タスク終了前の時間]タイプ)
- アクティビティを実行する制限時間に対する割合で指定([解決時間比]タイプ。[制限時間]タブページの[制限時間](SQL名:tsResolDelay)フィールドを使って計算)

💋 注意:

ワークフローの制限時間を定義する期間は、業務日で指定します。

タスクを作成するとすぐに、関連するワークフローのアラームが生成されます。 ワークフローの制限時間は、AssetCenter Serverで変更します。モニタの頻度は、 AssetCenter Serverのオプションで定義します。

### アラームの効果

アラームで次のインスタンスをトリガできます。

- AssetCenterで定義したアクション
- イベント。アラームによってトリガされるイベントは、[アラーム]タイプのイベントです。名前は、これらのイベントを定義したアラームと同じです。

# ワークフローの実行グループ

ワークフローの実行グループを使って、定義しているワークフローチャートを 分類できます。ワークフローチャートの実行グループは、[ツール/ワークフ ロー/ワークフローチャート]メニューを選択し、[全般]タブページの[実 行グループ](SQL名: GroupName)フィールドで指定します。

AssetCenter Serverは、新しいワークフロー実行グループの作成をモニタします。 AssetCenter Serverは、新しいワークフロー実行グループを検出し次第、[実行グ ループ 'G'のワークフロー規則を適用]という新しいモニタモジュールを作成 します。

この機能によって、次のことが可能になります。

- ワークフローの実行グループごとに検証スケジュールを定義できます。
- 異なるワークフローの実行グループごとに、別のAssetCenter Serverでモニタできる。

ワークフローの実行グループが検出されると、 AssetCenter Serverは実行グループのワークフローの規則(アラームのモニタ、[定期的]タイプのイベントの処理、トランジションの起動、タスクの実行など)をモニタし、実行します。

# ワークフローのトラッキング

AssetCenterのあるテーブルを、ワークフローチャートの開始オブジェクトのコ ンテキストとして定義すると、[ワークフロー]タブがそのテーブルの詳細画 面に表示されます。

この[**ワークフロー]**タブページには、選択しているレコードを開始オブジェ クトとして使う進行中のワークフローのステータスが表示されます。

各ワークフローは、それぞれの進行状況を示すサブタブページに表示されます。 • サブタブページの左側には、発生したイベントが一覧表示されます。

サブタブページの右側には、ワークフローチャートが表示されます。進行中のアクティビティが点滅表示されます。その後の手順は淡色表示されます。

# 終了済みワークフローインスタンスの削除

### 終了したワークフローインスタンスを削除する理由

ワークフローチャートを実行するとワークフローインスタンスが作成されます (**[ワークフローインスタンス]**(SQL名:amWfInstance)テーブル)。

これらのワークフローインスタンスは、実行終了後(**[ステータス]**フィール ド、SQL名:seStatus)でも自動的には削除されません。

ワークフローチャートを頻繁に使用する場合、作成されるワークフローインス タンスの数は莫大になります。

これはデータベースのサイズを無用に増大させるだけでなく、AssetCenterの性 能を低下させる原因にもなります。

このため、実行が終了したワークフローインスタンスを、定期的に削除するよ うに設定することをお勧めします。

## 終了済みワークフローインスタンスの自動削除

本番用データベースで、終了済みワークフローインスタンスの削除を自動化す るには、

AssetCenter

- [ワークフローインスタンス](SQL名:amWfScheme)テーブルにフィー ルドを1つ追加し、期限を指定できるようにします。これを使って、ワーク フローチャートの古いインスタンスが、一定の期限を経過した後に削除され るように指定します。
- 2 古いワークフローインスタンスを削除するアクションを作成します。
- 3 アクションを自動的に実行するワークフローチャートを作成します。
- 4 ワークフローチャートが自動的に実行されるように、AssetCenter Serverを設 定します。

### [ワークフローチャート]テーブルにフィールドを1つ追加 する

**[ワークフローチャート]**(SQL名:amWfScheme)テーブルに以下のフィール ドを追加します。

パラメータ	值
SQL名	AutoCleaningDelay
ラベル	インスタンス削除の期限
説明	この期限を過ぎると、終了済みのワークフローインスタ
	ンスは削除される。
データ型	期間
このフィールドのインデッ	このオプションは選択しません。
クスを作成	
説明( <b>[ヘルプの本文]</b> タ	<li>この期限を過ぎると、終了したワークフローインス</li>
ブ)	タンスは削除されます。
例	<ll>-1 : ワークフローインスタンスは決して削除されま</ll>
	せん。 <li>正の値またはNULL値:期限が過ぎるとワー</li>
	クフローインスタンスは削除されます。

既存のテーブルにフィールドを追加する方法については、マニュアル『管理』 の「データベースのカスタマイズ」の章、「新規オプジェクトの作成」の節の 「フィールド、リンクとインデックスの作成」を参照してください。

#### [インスタンス削除の期限]フィールドに入力する

使用する各ワークフローチャートの、**[インスタンス削除の期限]**(SQL名: AutoCleaningDelay)フィールドに値を入力します。

#### 本番用データベースに接続する

- 1 AssetCenterを起動します。
- 2 本番用データベースに接続します。

### 古いインスタンスを削除するアクションを作成する

1 アクションのテーブルを開きます。([ツール/アクション/編集])。

- 2 [新規作成]をクリックします。
- 3 以下のフィールドに値を入力します。

フィールド名	SQL <b>名</b>	値
名前	Name	終了済みワークフローインスタンスを削除す
		ත
コンテキスト	ContextTable	ワークフローチャート ( amWfScheme )
タイプ	seActionType	スクリプト
SQL名	SQLName	DeleteFinishedWfInstances
アクションのスクリ	Script	以下(*)参照
プト		

(\*)アクションのスクリプト

Const NumberOfInstanceToDelete = 50
Dim IRc As Long
Dim i As Long
i = 0
If [AutoCleaningDelay] >= 0 Then Dim hqWfInstance As Long hqWfInstance = AmQueryCreate() IRc = AmQueryExec(hqWfInstance, "SELECT IWfInstanceId FROM amWfInstan ce WHERE IWfSchId = "& [IWfSchId] & " And seStatus = 1 AND ADDSECOND S(dtCompleted, " & [AutoCleaningDelay] & ") < GetDate()" )
Do While (IRc = 0 And i < NumberOfInstanceToDelete) Dim hrWfInstance As Long hrWfInstance = AmGetRecordHandle(hqWfInstance) IRc = AmDeleteRecord(hrWfInstance) IRc = AmReleaseHandle(hrWfInstance) IRc = AmQueryNext(hqWfInstance) i = i + 1
Loop

4 [作成]をクリックします。

アクションを自動的に実行するワークフローチャートを作成 する

- ワークフローチャートのリストを表示します([ツール/ワークフロー/ ワークフローチャート]メニュー)。
- 2 [新規作成]をクリックします。
- 3 以下のフィールドに値を入力します。

フィールド名	SQL <b>名</b>	值
名前	Name	終了済みワークフローインスタンスの削除
参照番号	Ref	ADM_CLEAN_WF_INSTC
開始オブジェクトの コンテキスト	StartContextTable	ワークフローチャート(amWfScheme)
実行グループ	GroupName	AssetCenter Server内でワークフローチャート の実行を自動化できるように、名前を記入し ます(ADMINなど)。
1つのオブジェクト につき1つのワークフ ローを起動	bUniqueActive	このオプションを選択します。
データベースにイン スタンスを保存しな い	bTransient	このオプションは選択しません。

- 4 [作成]をクリックします。
- 5 [アクティビティ]タブページを選択します。
- 6 「**開始」**アクティビティで右クリックし、[イベントの追加]メニューを選択します。
- 7 以下のフィールドに値を入力します。

フィールド名	SQL <b>名</b>	値
名前	Name	Timer
タイプ	seMonitoringType	定期的
進行中のワークフ ローがある場合に ワークフローを再初 期化	bReinitialize	このオプションは選択しません。
AQL条件	AQLCond	AutoCleaningDelay >= 0

8 必要に応じて[パラメータ]タブにデータを入力します。

9 [追加]をクリックします。

10 右クリックし、[アクティビティの追加]メニューを選択します。 11 以下のフィールドに値を入力します。

フィールド名	SQL <b>名</b>	值
名前	Name	Clean W/F instances
タイプ	seType	自動アクション
タスクを保存	bLogWorkItem	このオプションを選択します。
コンテキスト	ContextTable	ワークフローチャート ( amWfScheme )
入力条件	seInCond	OR

- 12 [追加]をクリックします。
- 13 [パラメータ] タブページを選択します。
- 14 以下のフィールドに値を入力します。

フィールド名	SQL <b>名</b>	值
アクションを即実行	bExecImmediately	このオプションを選択します。
アクション	Actions	終了済みワークフローインスタンスを削除す る

- 15 **[閉じる]**をクリックします。
- 16 [Clean W/F instances] アクティビティをダブルクリックします。
- 17 [実行済]イベントを右クリックし、[イベントの詳細]メニューを選択します。
- 18 以下のフィールドに値を入力します。

フィールド名	SQL <b>名</b>	值
処理	seProcessingMode	イベントを保存してすぐに処理

- 19 [閉じる]をクリックします。
- 20 マウスを使って、 [**開始**] アクティビティと [Clean W/F instances] アクティ ビティ間にリンクを作成します。
- 21 [実行済]イベントを右クリックし、[終了イベント]メニューを選択します。
- 22 [変更]をクリックします。

### **ワークフローチャートが自動的に実行されるように、** AssetCenter Server**を設定する**

1 AssetCenter Serverを起動します。

AssetCenter

- 2 「終了済みワークフローインスタンスの削除」ワークフローの実行をトリガ するモジュールを設定します([ツール/モジュールの設定]メニュー)。 モジュールには「実行グループ'X'にワークフロー規則を適用」という名前 が付いています。「X」は、ワークフローチャートの[実行グループ](SQL 名:GroupName)フィールドで指定した値になります。
- 3 ワークフローを自動的に実行するには、AssetCenter Serverを稼動状態にして おきます。



本章では、AssetCenterデータの出力方法とデータベースのSQLビューの管理方法 について説明します。

# 定義

# 出力スクリプト

出力スクリプトを使うと、データの出力、AssetCenter Exportまたはamexp.exeを 使ったSQLビューの作成、再作成、または削除を行うことができます。出力ス クリプトを保存すると、後で繰り返し使用できます。

出力スクリプトは次の要素で構成されます。

- 出力クエリ。データベースから抽出するデータを定義します。
- データを出力するときの書式設定オプション
- SQLビューの作成および削除で実行するアクション

出力スクリプトは次のモードで動作します。

- データを出力する場合は「出力モード」
- データベースからSQLビューを作成(再作成)または削除する場合は「ビュー モード」

# 出力クエリ

出力クエリは、AssetCenter Exportを使って定義します。

出力クエリは、次の要素を使って定義します。

- 名前
- 生成される出力ファイル(「出力モード使用時」)
- コメント(出力されません)
- 開始テーブル
- 抽出する列のリスト(開始テーブルからのフィールド、リンク、任意管理項目、および特殊フィールド)および関連付けられている並べ替え基準
- WHERE句を含む、抽出条件を定義するフィルタ
- HAVING句を含む、抽出条件を定義するフィルタ
- クエリタブページ([フィルタ(WHERE句)]およびHAVING句の各タブページ)のテキスト
- プレビュータブページ

# AssetCenterデータベースからデータを出力する

AssetCenterデータベースからテキストファイルに、次の方法でデータを出力できます。

- 出力スクリプトを使う。
- [リストの出力]状況依存メニューを使う。このメニューは、少なくとも1 つのリストまたはタブページ内のリストが表示されている場合に表示されます。アクティブなリストを出力できます。

# 出力スクリプトを使ってデータを出力する

- 1 **[スタート]**メニューから、またはAssetCenterプログラムグループから AssetCenter Exportを起動します。
- 2 「出力モード」に設定した出力スクリプトを定義します。
  - 1 **[クエリ]**タブページで、出力するデータを定義するクエリを作成します。
  - 2 [**書式設定**]タブページで、データを出力するテキストファイルの書式 を指定します。
  - 3 [ファイル/保存]または[ファイル/名前を付けて保存]メニューを 使って、スクリプトを保存します。
- 3 次のいずれかの方法で出力スクリプトを実行します。
  - AssetCenter Exportで<スクリプトの実行> <sup>%</sup> アイコンから直接実行する。

• コマンドプロンプトからamexpl.exeを使う。

#### 💋 注意:

AssetCenterで定義したアクセス制限の整合性を維持する理由から、AssetCenter Exportを起動またはamexpl.exeを実行できるのは管理者(「Admin」ログインまたは管理者権限を持つユーザ)だけです。

# [リストの出力]状況依存メニューを使ってデータを 出力する

[リストの出力]状況依存メニューにはすべてのAssetCenterユーザがアクセス できます。ユーザは、自分の権限に応じてAssetCenterで表示できるデータのみ を出力できます。

- 1 出力するリスト(メインリストまたはタブページ内のリスト)を表示します。画面に複数のリストを表示している場合は、カーソルが出力するリスト内にあることを確認してください。
- 2 [リストの出力]ポップアップメニューを選択します。
- 3 表示されるウィンドウに入力し、[出力]ボタンをクリックします。

### 💋 注意:

[ファイル/リストの出力]メニューの詳細については、マニュアル『はじめ に』の「AssetCenterを初めて使用する」の章、「レコードリスト」の節の「リ ストを出力する」を参照してください。

## AssetCenterデータベースでSQLビューを管理する

AssetCenter Exportを使うと、AssetCenterデータベースでSQLビューを作成、再作成、または削除できます。その後、外部ツールでテキストファイルの代わりにこれらのビューを使用できます。

#### 💋 注意:

注意:出力スクリプトで作成、変更、および削除できるSQLビューは、AssetCenter で使われている意味でのビューとは異なります。SQLビューは、SQLの「CREATE VIEW」ステートメントに当たります。

AssetCenterデータベースでSQLビューを作成、再作成、または削除するには、次の手順に従います。

AssetCenter

- 1 AssetCenter Exportを起動します。
- 2 「ビューモード」に設定した出力スクリプトを定義します。
  - 1 **[クエリ]**タブページで、抽出するデータを定義するクエリを作成します。
  - 2 [ビュー]タブページで、実行するアクションを指定します。ビューの 作成、変更または削除、生成されるSQLスクリプトの直接実行、ファイ ルへの保存などのアクションを指定できます。
  - 3 出力スクリプトを保存します。
- 3 次のいずれかの方法で出力スクリプトを実行します。
  - AssetCenter Exportで直接実行する。
  - amexpl.exeを実行する。



出力したデータを再インポートする場合は、調整キーとしてテーブルの「Id」 フィールドを使わないようにしてください。実際には、対応するID番号は一定 ではなく、変更されることがあります。資産の資産タグなど、値が「変化しな い」キーを使います。

# 出力スクリプトを定義する

データの出力またはデータベースのSQLビューの作成を行うには、出力スクリ プトとその中で使う出力クエリを定義する必要があります。これを行うには、 AssetCenter Exportを使います。

出力スクリプトは次の要素で構成されます。

- 出力クエリ。データベースから抽出するデータを定義します。
- データを出力するときの書式設定オプション

SQLビューの作成および削除で実行するアクション

出力スクリプトは次のモードで動作します。

- データを出力する場合は「出力モード」
- データベースからSQLビューを作成(再作成)または削除する場合は「ビュー モード」

ここでは、出力スクリプトの作成方法について説明します。

- 出力スクリプトの作成方法
- 出力クエリを定義する
- 出力スクリプトの出力書式

• SQLビューに関するアクション

# 出力スクリプトの作成方法

出力スクリプトを作成または変更するには、次の手順に従います。

- 1 AssetCenter Exportを起動します。
- 2 適切なデータベースを開きます。注意:「Admin」ログインまたは管理者権 限を持つログインを使った場合のみ接続できます。
- 3 [ファイル/新規作成]メニューを使って新しいスクリプトを作成するか、 [ファイル/スクリプトを開く]メニューを使って変更するスクリプトを開きます。
- 4 AssetCenter Export画面の上部で、データを出力するか(出力モード)、デー タベースのSQLビューを管理するか(ビューモード)を定義します。
- 5 [**クエリ**]タブページで出力スクリプトのクエリを作成します。
- 6 データを出力する場合は、[**書式設定**]タブページで出力データの書式を指 定します。
- 7 SQLビューを管理する場合は、[ビュー]タブページで実行する操作を指定 します。
- 8 [ファイル/保存]メニューまたは[ファイル/名前を付けて保存]を使っ てスクリプトを保存します。

### 出力クエリを定義する

出力スクリプトのクエリは、AssetCenter Exportの**[クエリ]**タブページで定義 できます。

- 出力クエリを追加する場合は、[新規作成]ボタンをクリックします。
- 選択した出力クエリを削除するには、[削除]ボタンをクリックします。

## 出力スクリプトでクエリを作成する

- 1 [**クエリ**]タブページで[新規作成]をクリックします。
- 2 クエリ名を定義します。この名前は、出力スクリプトの詳細の[メッセー ジ]タブページの実行ログで使われます。
- 3 コメントを入力することは可能ですが、出力されません。
- 4 [**クエリ**]フィールドで、出力するデータを定義します。
- 5 ビューの作成、変更または削除ではなく、データーを出力する場合は、[ファ イル]フィールドに、クエリによって選択された出力データを書き込むテキ ストファイルのパスと名前を指定します。出力スクリプトに複数の出力クエ リが含まれている場合は、複数のテキストファイルが作成されます。

💋 注意:

[SQL**ビューの作成/破棄]**オプションを選択した場合は、[ファイル]フィー ルドは表示されません。

## 出力するデータ

出力するデータを指定するには、[クエリ]タブページの[クエリ]フィール ドに入力します。クエリは、AssetCenterデータベースのテーブルに適用されま す。

クエリは、直接入力するか、図ボタンをクリックしてクエリを簡単に定義でき るウィンドウにアクセスし、次のオプションを使って指定することもできます。

### 出力する列と並び順

[列/並べ替え]タブページでは、出力するフィールド、リンク、任意管理項 目、および特殊フィールドのリストと、その並べ替え基準を定義します。 左側のリストから出力に使うすべてのフィールド、リンク、任意管理項目、お よび特殊フィールドを1つずつ選択し、右矢印ボタンをクリックして、右側のリ ストに挿入します。

右側のリストの各列で次の操作を行います。

- 列を出力するには、[表示/非表示]チェックボックスをオンにします。
   [表示/非表示]チェックボックスをオンにしないと、列は出力されません
   (出力データの並べ替えなどに列を使用することは可能です)。
- [グループ化]ボックスをチェックして、列に対応するフィールドでデータ をグループ化します。これは、SQLクエリにGROUP BY <フィールド名>句 を追加するのと同じです。

例

SELECT Brand, Count(IModeIId) FROM amModel GROUP BY Brand ORDER B Y Brand

🌻 警告:

[**グループ化**] チェックボックスをオンにすると、クエリにはGROUPBYが 追加されますが、クエリを有効にするには、SELECT句に適切な集合関数を 追加する必要があります。

出力データの並び順を定義します。

 インデックスで並べ替えるには、[インデックスで並べ替え]フィールドで インデックスを選択します。  それ以外の場合は、並べ替える順番で項目の[並べ替え]チェックボックス をオンにします。

💋 注意:

SQL Anywhereを使う場合以外は、[インデックスの強制使用]オプションをオンにすると、クエリが参照するインデックスを強制的に使用することができます。詳しくは、『AssetCenterの高度な使い方』、「AQLクエリ」の章、「並べ替えとインデックス」の節を参照してください。

[固有のレコードのみ]チェックボックスをオンにすると、まったく同じ行が 複数ある場合は1度だけ出力されます。これは、SQLクエリにDISTINCT句を追 加するのと同じです。

[**固有のレコードのみ**]チェックボックスをオフにした場合のクエリ

SELECT Brand FROM amModel

[固有のレコードのみ]チェックボックスをオンにした場合のクエリ

SELECT DISTINCT Brand FROM amModel

#### フィルタ

次の2種類のフィルタを使って、抽出するデータを選択できます。

- [フィルタ(WHERE句)]タブページでWHERE句を使うAQLクエリ
- [HAVING句] タブページでHAVING句を使うAQLクエリ

#### クエリを表示する

[**列/並べ替え]、[フィルタ(WHERE句)]**、および[HAVING句]タブ ページで定義したAQLクエリは、[**クエリ**]タブページに表示されます。

### クエリ結果をプレビューする

クエリをテストし、その結果を**[プレビュー]**タブページにSQL言語のシンタッ クスで表示できます。

◎ をクリックするだけで、クエリ結果をレコードのリストとしてプレビューできます。ウィンドウの右下に、クエリに一致するレコード数が表示されます。

## 出力スクリプトの出力書式

[**データの出力**]オプションを選択した場合は、[**書式設定**]タブページで、 出力テキストファイルの書式を定義できます。この書式設定はすべての出力ク エリに適用されます。

#### 💋 注意:

ビューの削除、作成、再作成を選択した場合は、[**書式設定**]タブページは表示されません。

# 列タイトル

出力ファイルの1行目に含める要素を次の中から選択します。

- 出力クエリで指定した列のエイリアス
- 列に対応するフィールドまたはリンクの「SQL名」
- 列に対応するフィールドまたはリンクの「説明」

### 列の区切り

各列のデータ間に挿入する記号を指定します。

### テキスト識別子

テキスト文字列を囲む識別子を指定します。一重引用符(')を使うと、二重引 用符(")として出力されます。逆に二重引用符(")は、一重引用符(')とし て出力されます。

### 文字セット

ANSI、OEM(DOS)、UFT-8、UNICODEとLatin1文字セットのどれかを選択できます。

### 十進法の区切り

出力された数字の小数部を分離する文字を指定します。

### 日付の区切り

出力された日付の日、月、および年の間に挿入する文字を指定します。

### 日付の表示形式

出力される日(DD)、月(MM)、および年(YY)の順序を定義します。

### 西暦年の桁数

西暦年を2桁と4桁のどちらで出力するかを定義します。
#### 時刻の区切り

時間、分、および秒の間に挿入する文字を指定します。

#### 秒数表示

出力された時刻で秒を表示するかどうかを指定します。

## SQLビューに関するアクション

出力クエリに対応するSQLビューを削除または(再)作成する場合は、[**ビュー]** タブページを使って、実行するアクションを定義できます。

#### 💋 注意:

[**ビュー**]タブページは、[**データの出力**]オプションを選択した場合は表示 されません。

[アクション]オプションで、実行するアクションとして次のどちらかを選択します。

- ビューの作成または変更
- ビューの作成

「SQLビューの操作スクリプト」オプションで、クエリの処理方法を指定しま す(**[クエリ]**タブページ、「**アクション」**フレーム)。

- 出力スクリプトの実行時に直接SQLビューを作成または再作成するには、
   [直接SQLを実行]オプションを選択します。
- ビューを作成(CREATE VIEWステートメント)または削除(DROP VIEWス テートメント)するSQLビュースクリプトを作成するには、[SQLコードを ファイルに保存]オプションを選択してから、次の操作を実行します。
  - 1 回ボタンをクリックして、ファイルの名前とパスを指定します。
  - セミコロン(;)(Oracleの場合)と「GO」(他のすべてのDBMS)のどちらかをSQLステートメントの区切り文字として指定します。

## 出力スクリプトを実行する

出力スクリプトを使うと、データの出力やSQLビューの管理を行うことができ ます。

ここでは、出力スクリプトを実行する2つの方法について説明します。

- AssetCenter Exportから出力スクリプトを実行する
- DOSから出力スクリプトを実行する

## AssetCenter Exportから出力スクリプトを実行する

AssetCenter Exportから出力スクリプトを実行するには、次の手順に従います。 1 AssetCenter Exportを起動します。

- 2 出力スクリプトを定義し、保存します。
- 3 次のいずれかの方法でスクリプトを実行します。
  - [**アクション / スクリプトの実行**]メニューを使う。
  - [F8] キーを押す。
  - 🧏 をクリックする。

出力処理の進行状況は、[メッセージ]タブページに表示されます。 出力処理が正常に終了すると、最後に「スクリプトが問題なく実行されまし た。」というメッセージが表示されます。エラーが発生すると、「スクリプト を実行中にエラーが発生しました。」というメッセージが表示されます。 すべてのメッセージの前には、次のアイコンが表示されます。

- 一般情報
- エラー
- 問題なく出力
- ▲ 注意

## DOSから出力スクリプトを実行する

## 動作

DOSソフトウェアをオンラインで実行するには、AssetCenter Exportを使って、 事前に実行スクリプトを作成しておく必要があります。

その後、AssetCenterのインストール先ディレクトリのbinサブディレクトリにあるamexpl.exeプログラムを使って、手動でまたはバッチファイルを使って自動的に出力コマンドを実行できます。

## シンタックス

amexpl32 [-verbose] [-?|h|H] -script:<スクリプト名> -cnx:<cnx> [-login:<ログイン名>] [-password:<パスワード>]

-verbose:出力処理中にメッセージを表示します。 -?、-hまたは-H:ヘルプメッセージを表示します。 -script:実行する出力ファイルのパスと名前。

-cnx:AssetCenterデータベースへの接続名(**[ファイル/データベース接続の管理]**メニューに表示される名前)。 -login:データベース管理者のログイン名(「Admin」または管理者権限を持つ ユーザ)。 -password:ログインのパスワード <>内に入れる文字列にスペースを入れることはできません。 例 amexpl32 -verbose -script:biensibm.scx

-cnx:GeneraleDatabse -login:Gerald -password:PAssword



本章では、スクリプトの使用方法について説明します。

## スクリプトの定義



「スクリプト」という用語は、一般に高級言語で書かれたプログラムを指しま す。AssetCenterでは、次の3種類のスクリプトを使います。

- プロシージャスクリプト
  - フィールド値の計算、AssetCenterのデータベースオブジェクトのプロパ ティの条件付けなどに使われる、Basicで書かれた計算スクリプト
  - 特にアクションでのタスクを実行するBasicスクリプト

💋 注意:

これらのBasicプログラムには関数を組み込むことができます。ここでは 主にこのタイプのスクリプトについて説明します。

宣言スクリプト: Basicとは異なる独自のスクリプト言語を使うインポートおよび出力スクリプトです。このタイプのスクリプトについては、マニュアル

**『管理』の「インポート」**の章、および『AssetCenter**の高度な使い方』**の 「データの出力およびSQLビューの作成」を参照してください。

宣言スクリプトとプロシージャスクリプトの混合:このタイプのスクリプトは、AssetCenterのウィザードで使われます。

## 本バージョンのBasicに関する情報

AssetCenterで使われているBasicのバージョンは、Cypressによって開発されたサ ブセットで、Visual Basic for ApplicationsTMと互換性があります。この言語、構 造、およびシンタックスの詳細については、Basicのマニュアルを参照してくだ さい。

特定のVisual Basic for Applications関数だけをサポートしています。

- ファイルアクセス関数はサポートしていません。
- 日付および時刻関数のサポートに制限があります。
- Visual Basic for Applicationsコントロールは使用できません。

#### 💋 注意:

関数やBasicキーワードのプログラミングの参照を表示するには、調べる語の上 にカーソルを置き [F1] キーを押すと、ポップアップ メニューが表示されま す。

## データアクセス表記

AssetCenterで使われているBasicシンタックスは、現在のレコードからのデータ アクセス関数を除き、標準のシンタックスと似ています。データアクセスでは、 次の形式を使います。

[Link.Link.Field]

例:モデルのテーブルから

[Category.FullName]

#### 💋 注意:

次のシンタックスを使って、リンクのID番号を取得できます。

[Link.Link]

リンクを参照する場合は、リンクのSQL名またはリンクのキー名を使います。 例

AssetCenterの高度な使い方

RetVal=[Contact.Location] or RetVal=[Contact.ILocald] 両方の例とも、同じ結果、つまりリンクのIDが返されます。

## スクリプトの適用

AssetCenterでは、次のプロパティにBasicスクリプトを関連付けることができます。

- フィールドのデフォルト値の設定(ポップアップメニューの[オブジェクトの設定]コマンドを使って設定)
- テーブルに関連付けられた任意管理項目のデフォルト値
- Basicタイプの特殊フィールド
- フィールドの設定(ポップアップメニューの[オブジェクトの設定]コマンドまたはAssetCenter Database Administratorを使って設定)
  - デフォルト値
  - 必須属性
  - 履歴
  - 読取り専用
- テーブルに関連付けられている任意管理項目のパラメータ
  - [デフォルト値] (SQL名: DefValScript)
  - [使用可能] (SQL名: seAvailable)
  - [デフォルトで表示] (SQL名: seForceDisplay)
  - [必須] (SQL名: seMandatory)
  - [履歴の保持] (SQL名: seKeepHistory)
- [スクリプト]アクション
  - [スクリプト]アクションの[アクションのスクリプト](SQL名: Script)
- ウィザード
  - ウィザードスクリプトの開始および終了
  - ノードプロパティを定義するスクリプト
- Basicタイプの特殊フィールド
- ワークフロー
  - [**テスト/スクリプト**]ワークフローのアクティビティ
  - [データペース] ワークフローのイベント
  - 「計算された」ワークフローの担当者

## 関数の概要

ここでは、次の内容について説明します。

- 関数の定義
- 組み込み関数とプログラマブル関数
- 関数とパラメータのデータ型

#### 関数の定義

「関数」とは、なんらかの処理を実行し、値をユーザに返すプログラムです。 返される値を「戻り値」または「戻りコード」といいます。 関数は次のような構造になっています。

Function <関数名> (<パラメータ> As <パラメータの種類>[, ..., <パラメータ> As <パラメータの種類>]) As <関数の種類>

<関数によって実行されるプログラム(スクリプト)。このプログラムで戻り値 を定義します>

End Function

この構造は、組み込み関数とプログラマブル関数の両方に当てはまります。

## 組み込み関数とプログラマブル関数

AssetCenterでは、主に組み込み関数とプログラマブル関数を使うことができます。

#### 組み込み関数

組み込み関数は、ユーザ用に事前に作成されたソフトウェアアイテムに似てい ます。これらのソフトウェアアイテムは、すべてのタイプのタスク(ユーザか ら要求されるデータの計算、変換)を実行し、結果を返します。ユーザは、名 前を使って関数を呼び出し、結果を得るために必要な情報を指定します。ユー ザが指定する情報項目を「パラメータ」と呼びます。

例えば、AmConvertCurrency()関数は、特定の日付に定義された為替レートを使って、通貨Aの金額を通貨Bの金額に換算します。この場合は、次のような情報が必要になります。

- 関数名はAmConvertCurrency
- ユーザが指定するパラメータ
  - 通貨A

AssetCenterの高度な使い方

- 通貨B
- 換算する金額
- 換算を実行する日付(使用する為替レートの識別に使います)

この関数は換算を実行し、換算結果に対応する戻り値をユーザに返します。

#### プログラマブル関数

プログラマブル関数は、ユーザ自身が作成できるソフトウェアアイテムです。 ユーザは、プログラマブル関数によってRetVal変数(「戻り値」とも呼ばれる) に戻される値を、次の形式で明示的に定義する必要があります。

RetVal=<式>

#### 💋 注意:

AssetCenterでは、戻り値が定義されていない関数のスクリプトはコンパイルできません。

プログラマブル関数には、スクリプトビルダからアクセスできます(スクリプ トを作成できるフィールドの図ボタンをクリックすると表示されます)。スク リプトビルダは、関数に対応するソフトウェアアイテムを、ユーザが簡単に作 成できるように設計されています。スクリプトビルダには、プログラマブル関 数を作成するためのテンプレートがあります。 AssetCenter

#### 図 10.1. スクリプトビルダ



プログラマブル関数の説明は、スクリプトビルダウィンドウの上部に表示され ます。関数に関連するオブジェクト(例えば、資産テーブルの[パーコード] (SQL名: BarCode)フィールドのデフォルト値)と、予測される戻りコード (前の例では「文字列」)を確認できます。

## 関数とパラメータのデータ型

## 関数のデータ型

組み込み関数のデータ型は、関数から戻される値のデータ型になります。Basic スクリプトでは、コンパイルエラーや実行エラーの原因となることがあるので、 この点によく注意してください。

例えば、あるデータ型のフィールドのデフォルト値を定義しているときに、別 のデータ型の値を戻す関数は使用できません。例えば、下のデフォルトスクリ プトに「日付」または「日付+時刻」型のフィールドを割り当てようとしたと します。

RetVal=AmLoginName()

AmLoginName() 関数は接続ユーザ名を文字列で返します(文字列型)。そのため、この場合の戻り値の形式は、「日付」フィールドの形式とは互換性がない

ので、次に同じテーブルでレコードを作成すると、エラーメッセージが表示されます。

#### パラメータのデータ型

組込み関数で使われるパラメータにもデータ型があります。実行する関数に適したデータ型を使う必要があります。パラメータのデータ型にエラーがある場合は、関数の実行時にエラーメッセージが表示されます。

#### 使用可能なデータ型

関数とパラメータで使えるデータ型とその説明は、次の表の通りです。

#### 表 10.1. 使用可能なデータ型

タイプ	説明
Integer(整数)	-32,768から+32,767の整数
Long(倍長整数)	-2,147,483,647から+2,147,483,646の整数
Double(倍精度)	8バイト浮動小数点数
String ( 文字列 )	任意の文字からなるテキスト
日付	日付または日付+時刻
Variant(可変)	汎用の任意の型

## プログラマブル関数の戻り値のデータ型を決定する

スクリプトを編集する前に、使用する関数とそのデータ型を決定する必要があ ります。この情報は、すべての「BASICスクリプト」ウィンドウに次の形式で 太字で表示されます。

Function <関数名>() As <関数のデータ型>

最も一般的な関数のデータ型は、「ブール」、「整数」、および「文字列」で す。

- 「ブール」 関数は、TRUE(真)またはFALSE(偽)を返します。他の値は、 Basicスクリプトのコンパイル時にエラーの原因となります。
- 「整数」関数は、整数値だけを返します(0、1、8、12など)。
- 「文字列」 関数は、引用符で囲まれた文字列だけを返します("Building21"など)。

#### 💋 注意:

関数のデータ型に従わない場合は、Basicプログラムのコンパイル中にエラーが 発生します。常に使用する関数のデータ型に注意してください。

関数名とデータ型を使って、スクリプトで使う戻りコードを次の形式で指定で きます。

RetVal=<関数のデータ型に従った式>

## Basic**関数を分類する**

スクリプトで使われているBasicでは、様々な種類の関数が使われます。

- Visual Basic for Applications TM標準に従った従来のBasic 関数
- AssetCenterに固有の汎用関数。スクリプトを使う機能で使用できます。
- 専用関数。AssetCenterの特定の機能で使用できます。

## スクリプト作成の基本

ここでは、スクリプトの機能について、事例を紹介しながら説明します。

- スクリプト例の内容
- 手順1:任意管理項目「チュートリアル」を作成する
- 手順2:編集ウィンドウを開く
- 手順3:アルゴリズムを分析および定義する
- 手順4: Basicスクリプトを作成する
- 手順5: Basicスクリプトをテストする

## スクリプト例の内容

#### 目的

「チュートリアル」任意管理項目が「コンピュータ/マザーボード/」モデルだけ で使用できることを確認します。

#### 方法

「チュートリアル」任意管理項目の**[使用可能]**(SQL名:seAvailable)パラ メータにBasicスクリプトをリンクします。

## 手順1:任意管理項目「チュートリアル」を作成する

[管理/任意管理項目]メニューを選択します。[新規作成]をクリックして、 新しい任意管理項目を作成します。この任意管理項目に次のように入力します。

フィールド名	フィールドの値
[タイトル] (SQL: TextLabel)	チュートリアル
<b>[</b> SQL <b>名]</b> (SQL名:SQLName)	チュートリアル
<b>[入力タイプ]</b> (SQL名:seDataType)	[数値]

[作成]をクリックして、任意管理項目を作成します。

[**パラメータ**]タブページに移動し、 ■をクリックして、「チュートリアル」 任意管理項目のパラメータを編集します。 [制限]タブページに次のように入 力します。

■ 任意管理項目パラメータチュー	トリアル (Tutorial) (amModel)'の詳細		<u>_                                    </u>
<mark>テーブル:</mark> <u></u> モデル (	m Model)	-	新規作成( <u>N</u> )
任意管理項目: チュートリアノ	/ (Tutorial)	Q E -	複製(∐)
制限(C) デフォルト値(V) 入力	ヘルブ(円) 履歴		
使用可能:	(スクリプト)	• ٩	
デフォルトで表示:	いいえ	• 9,	
必須:	いいえ	• 9,	
履歴の保持:	いいえ	• 94	
□ メインレコードの作成中でも履	健歴を保持		閉じる(©)

## 手順2:編集ウィンドウを開く

[制限]タブページ、[使用可能](SQL名:seAvailable)パラメータを「(ス クリプト)」に設定します。国ボタンをクリックします。スクリプト編集ウィ ンドウが開きます。

## 手順3:アルゴリズムを分析および定義する

次のタスクを実行するアルゴリズムを作成する必要があります。

- モデルが「/コンピュータ/マザーボード」または従属モデルの場合に、[使用可能](SQL名: seAvailable)フィールドを[はい]に設定する。
- それ以外の場合はすべて、[使用可能](SQL名: seAvailable)フィールド を[いいえ]に設定する。

アルゴリズムは次のようになります。

If the model's full name starts with "/コンピュータ/マザーボード/" Then The feature is available Otherwise The feature is not available

つまり、任意管理項目の**[使用可能]**(SQL名: seAvailable)フィールドの値 は、モデルのテーブルの**[完全名]**(SQL名: FullName)フィールドの値によっ て決まります。アルゴリズムでは、このフィールドしか使いません。

[使用可能](SQL名: seAvailable)フィールドの横の図ボタンをクリックして、Basicスクリプトの編集を開始します。編集ウィンドウのドロップダウンリストから、モデルのテーブルから[完全名](SQL名:FullName)フィールドを選択できます。

フィールドを選択したら、 **・** ボタンをクリックして、編集ウィンドウに値を 移します。

## 手順4:Basicスクリプトを作成する

Basicスクリプト作成用のウィンドウで、スクリプトを作成します。このスクリ プトは、手順3で作成したBasicアルゴリズムを変換したものです。

「Left([FullName], Len("/コンピュータ/マザーボード/"))="/コンピュータ/マザー
<b>ボー ド/" Then</b>
RetVal=1
lse
RetVal=0
nd If

💋 注意:

スクリプトでは、大文字と小文字は区別されません。

[OK]をクリックして、スクリプトの作成を確定します。

## 手順5:Basicスクリプトをテストする

この手順では、スクリプトが正常に機能することを確認します。

- 1 [ポートフォリオ/モデル]メニューを選択して、モデルのテーブルを開き ます。[新規作成]をクリックして、新しいモデルを作成します。
- 2 必須フィールドだけに入力します。

1 [名前]

- 2 **[親モデル]**(SQL名: Parent)フィールドに「コンピュータ/マザーボード」と入力します。
- 3 [属性] (SQL名: Nature)
- 4 [バーコード] (SQL名: BarCode)
- 3 [作成]をクリックして、この新しいモデルを作成します。
- 4 [任意管理項目]タブページを表示し、■ボタンをクリックして任意管理項 目を追加します。選択画面に、スクリプトを編集した任意管理項目の名前が 表示されます。
- 5 [親モデル]の値を「/コンピュータ/」に変更します。[変更]をクリック して変更を確定します。
- 6 モデルの詳細画面の[任意管理項目]タブページを選択し、■をクリックして任意管理項目を追加します。選択画面には、スクリプトを編集した任意管理項目の名前が表示されません。

このスクリプトは正常に機能していることになります。

## スクリプトライブラリ

AssetCenterでは、スクリプトへのアクセスを統括するために、スクリプトライ ブラリを保存できるようになっています。

[管理/スクリプトライブラリ]メニューを使うと、スクリプトライブラリに アクセスできます。

保存されたライブラリはAPIコマンドamEvalScriptにより呼び出されます。

APIのamEvalScriptに関する詳細は、『プログラマーズリファレンス』の「関数の説明」の節を参照してください。

## 概念

AssetCenterでは、1つのスクリプトは1つの関数を定義します。 スクリプトライブラリを作成するということは、複数の関数の集まりを定義す ることに当たります。

## スクリプトライブラリを作成する

スクリプトライブラリを作成するには、次の手順に従います。

- 1 スクリプトライブラリのリストを開きます。
- 2 [名前]フィールドにライブラリの名前を入力します。
- 3 [**スクリプト**]フィールドにスクリプトを入力します。

4 [作成]をクリックして作成を確定します。

例えば、「biblio」という名のライブラリを作成し以下のスクリプトを入力します。

function FullName(strName As String, strFirstName As String) As String FullName = strFirstName & ", " & strName

end function

この関数は、従業員の姓と名から構成される文字列を戻します。

#### 🌻 警告:

各関数の名前は、複数のスクリプトライブラリ全体の中で、固有の名前でなければなりません。

## スクリプトライブラリ内に記録されているスクリプト を呼び出す

ライブラリからスクリプトを呼び出すには、ライブラリの名前、スクリプト内 で定義された関数と、関数に関連付けられたパラメータを定義する必要があり ます。

例えば、先ほど作成したライブラリを使用するスクリプトタイプの、「callEvalScript」アクションを作成するとします。

- 1 **[コンテキスト]**(SQL名:ContextTable)フィールドに**[部署と従業員]** (SQL名:amEmplDept)テーブルを入力します。
- 2 [スクリプト]タブに以下のスクリプトを入力します。

Dim strFullName As String strFullName = amEvalScript("biblio", "FullName", "", [Name], [FirstName]) amMsgBox (strFullName)

このスクリプトは「biblio」ライブラリから「FullName」関数を呼び出し、 ダイアログボックス内に従業員の姓名を表示します。

3 **[作成]**をクリックして作成を確定します。

#### 💋 注意:

通常「amEvalSript」APIと共に使用されるコンテキストパラメータは、スクリプトライブラリを呼び出す場合には使用されません。

## ヒントと注意事項

ここでは、スクリプトを作成する時に役立ついくつかのヒントを紹介します。

## プログラマブル関数を使うときの注意事項

スクリプトを作成するときは、次の点に注意してください。

- フィールドやリンクのデフォルト値を定義する関数などのプログラマブル関数は、関数の戻り値を設定することが目的です。プログラマブル関数内で他の操作を実行しないでください。関数が正常に機能しているときは一般的にパフォーマンスが向上しますが、関数を誤って使うとデータベースが破損するおそれがあります。
- プログラマブル関数は、AssetCenterの様々な機能で使用します。可能な限り スクリプトを最適化して、AssetCenterの全体的なパフォーマンスを維持して ください。

## スクリプトで使用する「日付+時刻」定数の形式

スクリプトで参照する日付は、ユーザが定義した表示形式に関係なく、次の国際標準形式で指定します。

yyyy/mm/dd hh:mm:ss

例

RetVal="2001/07/12 13:05:00"

#### 💋 注意:

日付の区切り文字としてハイフン (-)を使うこともできます。

## Basicでの日付形式

Basicでは、国際標準形式、またはDouble(倍精度)型の浮動小数点数で記述します。浮動小数点数の場合は、整数部は1899年12月30日の午前0時から数えて現在まで経過した日数、小数部は本日の午前0時から現在まで経過した時間の1日(86400秒)に対する割合(現在までの経過秒数を86400で割ったもの)を示します。

## スクリプトでの「Duration (時間)」の形式

スクリプトでは、時間を秒単位で記述して保存します。例えば、Duration(時間)型フィールドのデフォルト値を3日に設定するには、次のスクリプトを使います。

RetVal=259200

同様に、AmWorkTimeSpanBetween()などの時間を計算する関数も、秒単位の時間を返します。

💋 注意:

AssetCenterでは、1年は12ヶ月、1月は30日として計算するので、1年は360日になります。

## システムリストデータの読取りと書込み

AssetCenterでは、リストデータ内のそれぞれの値に整数を割り当てて、システムリストデータを管理しています。

資産の詳細画面の**[全般]**タブページにある**[標準割当]**枠内の**[割当]**(SQL 名:seAssignment)フィールド用の、リストデータの例は以下の通りです。

リストデータの値に対応する整数は次のようになります。

リストデータの値	対応する整数
[使用中]	0
[在庫中]	1
[除却済]	2
[納品待ち]	3

この原理から、リストデータのデフォルト値を定義するには、次の操作を行う 必要があります。

1 リストデータの値に対応する整数を確認する。

2 次の文字列を編集する。

RetVal=<リストデータ値に対応する整数>

例えば、[割当]フィールドで使うシステムリストデータのデフォルト値を[納 品待ち]に設定するには、文字列を次のように編集する必要があります。

RetVal=3

#### 💋 注意:

システムリストデータを、ユーザ定義のクローズドリストデータと混同しない でください。

#### 💋 注意:

システムリストデータ値の完全なリストは、AssetCenterのインストール先フォ ルダ内にある「doc/infos」サブフォルダの「Database.txt」にあります。「Data display and entry type」と「Additional information on data display and entry type」と いう2つの列に、リストデータのデータ型と各リストデータの値の一覧がありま す。

## CurrentUser**仮想リンク**

#### 定義

CurrentUserは、先ず全テーブルを検索し、次に[部署と従業員]テーブル内に ある現在のユーザのレコードに検索を絞るリンク、と見なされます。

- 「CurrentUser」という形式を使うと、現在のユーザのレコードを検索し、 ユーザのID番号を返します。
- 「CurrentUser.<フィールドのSQL名>」という形式を使うと、現在のユーザの フィールドの値を返します。

#### 💋 注意:

仮想リンクは、フィールドとリンクのリストには表示されません。そのため、 CurrentUserは、AssetCenter内のスクリプトビルダには直接表示されません。手 動で「CurrentUser」と入力してください。

## 等価関数

AmLoginName()とAmLoginId() 関数は、それぞれ現在のユーザ名とIDを返す関数で、CurrentUserから派生した関数と見なされます。これらの関数には、次のような関係が成り立ちます。

- AmLoginName()=[CurrentUser.Name]
- AmLoginId()=[CurrentUser.IPersId]

## Basic スクリプトにコメントを付ける

Basicスクリプトにコメントを付けスクリプトの機能を説明すると、ユーザのス クリプトの理解が深まり、ユーザが各自でスクリプトを変更できるようになり ます。AssetCenterでは、一重垂直引用符(')文字を使って、スクリプトの本体 にコメントを付けることができます。一重垂直引用符(')で始まる行の文字は すべて、コンパイラによってコメントとして解釈され、無視されます。コメン トを付けるには、次の2つの方法があります。

Basicスクリプト内にコメントだけの行を加える。特に注意事項はありません。

| 資産のテーブルのBarCodeフィールドの値をテストする | この値がPC1の場合は戻りコードをTRUEに設定する |f [BarCode]="PC1" Then RetVal=True End If

スクリプト行の終わりにコメントを追加し、Basicコンパイラにコメントとして解釈させる。この場合は、コロン(:)文字を使ってコメントとスクリプトを分離する必要があります。コメントの前に一重垂直引用符(')を付けます。

If [BarCode]="PC1" Then : ' BarCodeがPC1の場合 RetVal=TRUE : ' 戻り値をTRUEに設定 End If : ' テストの終わり

## エラーメッセージをトリガする

Err.Raise関数を使って、必要に応じてエラーメッセージをトリガすることができます。シンタックスは次の通りです。

Err.Raise (< Error number>, < Error message>)

#### 💋 注意:

ユーザがテーブルで作成または変更したレコードが、AssetCenter Database Administratorの[**有効**]フィールドの値により無効になった場合などに、Err.Raise 関数を使ってエラーメッセージをトリガさせると便利です。警告メッセージを 表示させることにより、レコードが作成または変更できない原因をユーザに知 らせることができます。

## スクリプト例1

ここでは、Basicスクリプトを使って処理できる作業例を示します。スクリプトの記述方法の説明を読む前に、自分でスクリプトを検討してみてください。

## スクリプトの目的

作業指示のテーブルに関連付けられている「例1」という任意管理項目は、作業 指示がクローズした時に入力する必要があります。作業指示がオープンの時は、 入力しなくても構いません。この例ではこの任意管理項目は作成済みで、下図 のようにリンク先のテーブルは作業指示であり、[使用可能]と[デフォルト で表示]が制限パラメータとして設定されていることにします。

№ 任意管理項目パラメータ例1(	Example_1) (amWorkOrder)'の詳細		_ 🗆 ×
テーブル: III 作業指示 (amWorkOrder) ・		•	新規作成(N)
任意管理項目: 例1 (Examp	le_1)	۹ 🗉 🔻	複製(型)
制限(C) デフォルト値(V)	「ヘルゴ(日) 履歴		
使用可能:	ולני	• 9,	
デフォルトで表示:	เสนา	• 9,	
必須:	(スクリプト)	• Q	
履歴の保持:	เหม	• 9	
□ メインレコードの作成中でも ■	夏歴を保持		閉じる( <u>C</u> )

## 手順1:アルゴリズムを分析および定義する

次のタスクを実行するアルゴリズムを作成する必要があります。

- チケットがクローズドの場合、[必須](SQL名: seMandatory)フィールド を[はい]に設定する。
- それ以外の場合はすべて、[必須](SQL名: seMandatory)フィールドを [いいえ]に設定する。

アルゴリズムは次のようになります。

If the work order is closed then Populating the feature is mandatory Else Populating the feature is not mandatory

作業指示の[ステータス](SQL名: seStatus)フィールドが[クローズド]の 場合、作業指示は終了しています。

したがって、任意管理項目の[必須](SQL名:seMandatory)フィールドの値 の条件となるのは、作業指示テーブルの[ステータス](SQL名:seStatus) フィールドの値です。アルゴリズムでは、このフィールドしか使いません。 [BASICスクリプト]画面のドロップダウンリストから、作業指示テーブルの [ステータス]フィールドを選択します。

フィールドを選択したら、 🔸 ボタンをクリックして、編集ウィンドウに値を 移します。

このフィールドには、システムリストデータの値が入ります。

# リストデータの値 対応する整数 [報告済] 0 [作業日確定済] 1 [進行中] 2 [クローズド] 3

ここでは以下のリストデータの値を使用します。

• 値「3」に対応する[**クローズド**]

## **手順2:**Basic**スクリプトを書く**

手順1のアルゴリズムをBasicで作成します。

🚱 BASICスクリプト	
'必須:テーブル'作業指示 (amWorkOrder)' 戻り値の型= 'Integer'	*
	<u>~</u>
■ フィールド(F): seStatus	• •
Function RetVal() As Integer	
[seStatus]RetVal = 3 then	
Potval – 1	
Else	
Retval = 0	
End If	-
End Function	
ОК	キャンセル

[OK]をクリックして、スクリプトの作成を確定します。

## 手順3:Basicスクリプトをテストする

この手順では、スクリプトが正常に機能することを確認します。

- 1 [ポートフォリオ/作業指示]メニューを使って[作業指示]テーブルを表示し、ステータスが[クローズド]の作業指示を選択します(ない場合は、 このステータスの作業指示を作成します)。
- 2 [任意管理項目]タブページを選択します。関連する作業指示には任意管理 項目が追加され、この任意管理項目は必須になっています。
- 3 ステータスが[クローズド]以外の作業指示を選択します。この作業指示の [任意管理項目]タブページに移動します。任意管理項目「例1」は表示されますが、この任意管理項目はオプションフィールドになっています。

このスクリプトは正常に機能していることになります。

## **スクリプト例**2

ここでは、Basicスクリプトを使って処理できる作業例を示します。スクリプトの記述方法の説明を読む前に、自分でスクリプトを検討してみてください。

## スクリプトの目的

従業員の詳細画面の[フィールド1](SQL名: Field1)フィールドに、従業員 の名前と姓がわかる場合はその両方を、名前がわからない場合は姓だけをデフォ ルトで表示するようにします。

## 手順1:アルゴリズムを分析および定義する

次のタスクを実行するアルゴリズムを作成する必要があります。

- 従業員の姓と名前の両方がわかる場合は、従業員の詳細画面の[フィールド 1](SQL名: Field1)に従業員の姓名をデフォルトで表示する。
- 名前がわからない場合は、従業員の詳細画面の[フィールド1](SQL名: Field1)に姓だけをデフォルトで表示する。

アルゴリズムは次のようになります。

If the first name of the employee does not exist then The default value of "Field1" is the name of the employee Else The default value of "Field1" is "Name, First"

従って、従業員の詳細画面の**[フィールド1]**(SQL名:Field1)のデフォルト 値の条件となるのは、従業員のテーブルの**[部署名/姓]**(SQL名:Name)フィー ルドと**[名前]**(SQL名:FirstName)フィールドの値です。アルゴリズムでは、 これら2つのフィールドしか使いません。

従業員の詳細画面の[フィールド1](SQL名:Field1)をマウスの右ボタンで クリックして、ポップアップメニューから[オブジェクトの設定]を選択しま す。

[**デフォルト]**フィールドの横にある 国ボタンをクリックして、Basicスクリプトを編集します。

## 手順2:Basicスクリプトを書く

手順1のアルゴリズムをBasicで作成します。

● BASICスクリプト         _□ ×           'デフォルト値: フィールド'フィールド1 (Field1)'(テーブル'従 ▲           業員 (am EmplDept)'内)           戻り値の型= 'String'
□ フィールド(E): Name ・
Function RetVal() As String If [Firstname] = " " Then Retval = [Name]
Else Retval = [Name] +" , "+ [Firstname] End If
End Function OK キャンセル

[OK]をクリックして、スクリプトの作成を確定します。

## 手順3:Basicスクリプトをテストする

この手順では、スクリプトが正常に機能することを確認します。

- 1 [ポートフォリオ/部署と従業員]メニューを選択して[部署と従業員] テーブルを開き、新しい従業員を作成します。
- 2 [部署名/姓](SQL名:Name)フィールドと[名前](SQL名:FirstName) フィールドに入力し、[作成]をクリックして作成を確定します。[フィー ルド1](SQL名:Field1)に従業員の姓と名前が表示されます。

このスクリプトは正常に機能していることになります。



カレンダのリストにアクセスするには、**[ツール/カレンダ]**メニューを使います。

## カレンダの概要

## カレンダの詳細画面

カレンダの詳細画面には、次の情報が表示されます。

- カレンダを固有に識別できる一般情報
  - カレンダの [名前] (SQL名: Name)
  - カレンダがリンクしている [タイムゾーン] (SQL名: TimeZone)
- [予定表]タブページ:通常の業務時間
- [例外]タブページ: [予定表]タブページで指定した予定表を適用しない 期間のリスト
- [予定表]および[例外]タブページで定義した規則を適用した業務時間の プレビュー

## カレンダの使用方法

カレンダは次の項目に関連付けられます。

• ワークフローのアクティビティ

ワークフローのアクティビティで定義したアラームをいつトリガするかを設定 できます。これらのアラームで定義する**[制限時間]**は、実際の業務時間で指 定します。

#### 🌻 警告:

データベースでカレンダを変更すると、ユーザがAssetCenterを終了し再起動した時に、カレンダにリンクしているフィールドにのみ変更が適用されます。

## カレンダに影響される機能

カレンダは、AssetCenterの特定の機能に影響を与えます。カレンダを変更すると、データベースの特定のレコードのデータが直接および間接的に変更されま

- す。カレンダは次の要素に関連しています。
- ワークフローのタスクの実行時間
- ワークフローのアクティビティに関連付けられているアラーム

## カレンダの作成方法

次の手順でカレンダを作成します。

- 1 カレンダに[名前] (SQL名: Name)を付けます
- 必要に応じて、[タイムゾーン](SQL名:TimeZone)フィールドに入力して、カレンダをタイムゾーンに関連付けます。
- 3 カレンダの詳細画面の[予定表]タブページで、通常の業務時間を定義しま す。
- 4 カレンダの詳細画面の[**例外**]タブページで、これらの業務時間以外の予定 表を定義します。
- 5 最後に、[プレビュー]タブページでカレンダの機能を確認できます。

## カレンダの作成手順

カレンダの作成手順は以下の通りです。

AssetCenterの高度な使い方

- 1 一般情報を入力する
- 2 [予定表]タブページに入力する
- 3 [除外]タブページに入力する
- 4 カレンダをチェックする

## 一般情報を入力する

業務時間と除外項目の詳細情報を入力する前に、詳細画面の[名前](SQL名: Name)フィールドに、カレンダを識別する名前を入力します。

また、**[タイムゾーン]**(SQL名:TimeZone)フィールドに、カレンダに使う タイムゾーンを指定することもできます。

## [予定表]タブページに入力する

カレンダの詳細画面の[予定表]タブページでは、カレンダの週単位の予定表 を定義します。このタブページで定義した業務時間によって週単位の予定表が 決まります。業務時間には一般的な規則を定義します。休日などはこの規則の 適用外として、[例外]タブページで定義します。

 予定表()
 例外(2)
 ブレビュー(2)

 週間予定表の定義
 0.3
 6
 9
 12
 15
 18
 21
 2

 月曜日(2)
 9.00-12:00;13:00-18:00
 9:00-12:00;13:00-18:00
 9:00-12:00;13:00-18:00
 9:00-12:00;13:00-18:00

 水曜日(2)
 9:00-12:00;13:00-18:00
 9:00-12:00;13:00-18:00
 9:00-12:00;13:00-18:00

 全曜日(2)
 9:00-12:00;13:00-18:00
 9:00-12:00;13:00-18:00
 9:00-12:00;13:00-18:00

 土曜日(2)
 9:00-12:00;13:00-18:00
 9:00-12:00;13:00-18:00
 9:00-12:00;13:00-18:00

#### 図 11.1. カレンダの詳細画面の [予定表] タブページ

各曜日ごとに、複数の業務時間を定義できます。次の2つの方法で定義できま す。

- 週の各曜日を表す目盛スライダーコントロールを使って定義する。
  - 1 業務時間の開始点でコントロールをクリックします。

- 2 業務時間の終点までマウスをドラッグします。目盛コントロールの右側 のテキストボックスに自動的に時間が入力されます。
- 3 必要に応じて、以上の操作を繰り返します。
- テキストフィールドに業務時間を直接数値で指定する。次のシンタックスで 入力します。

<業務時間の開始時間>-<業務時間の終了時間>;<業務時間の開始時間>-<業務時 間の終了時間>;…

時刻は次の形式で入力します。

<hh:mm[{午前|午後}]>

オプションの[午前|午後]パラメータを定義しない場合は、24時間形式を使うものと見なされます。

テキストフィールドの左側の目盛スライダコントロールに自動的に入力されます。

💋 注意:

グラフィカルコントロールを使う場合は、30分単位で指定できます。直接時間 を数値で入力する場合は、1分単位で指定できます。

## [除外]タブページに入力する

カレンダの詳細画面の[**例外**]タブページでは、[**予定表**]タブページで定義 した週単位の業務時間を適用しない期間を定義します。

### 出力スクリプトの作成方法

例外の期間の作成方法は次の通りです。

- 1 例外の期間に[名前]を付けます。
- 2 **[日数]**フィールドに、業務時間を適用しない日数を指定します。日数は 日、月、または年単位で定義できます。
- 3 [作業時間]フィールドを使って、例外期間中の業務時間を定義することもできます。このフィールドを使うと、「各月の最後の金曜日、チームは午前8時30分から午前10時30分までと、午後5時30分から午後6時30分まで勤務する」など、より細かく期間を定義できます。

[例外]タブページ

■ カレンダ 「標準力	レンダ - 日本' の詳細	<u>_   ×</u>
名前:	標準カレンダ - 日本	新規作成( <u>N</u> )
SQL名:	japanese_calendar	複製(∐)
タイムゾーン:	× E	•
予定表①	例外④ プレビュー◎	
✓ 名前 成人の日 成人の日 海の日 天皇誕生日 国民の休日 国体育の日 敬老の日 子供の日	▲ 日数 ▲ 葉務時間 2001/1/8 2003/1/13 */7/20 */12/23 */5/4 */10/10 */9/15 */5/5 */5/5	
日業務時	○日     ○月     ○年       鼓①<日付指定	閉じる( <u>C</u> )

このタブページは、次の2つのセクションで構成されています。

- タブページの上部には例外期間のリストが表示されます。ツールバーのボタンを使って、例外期間の作成、複製、破棄、変更および取り消しを行うことができます。
  - 🎦 <sup>і</sup> 新しい例外期間を作成します。
  - 🎽 例外期間を削除します。
  - 🖣 例外期間を複製します。

💋 注意:

[ランク]列を使うと、例外期間を優先順位で並べ替えることがきます。同 じ指定の例外期間がある場合に、優先順位を確認できます。例外期間には自 動的にランク(P00からP15)が割り当てられます。数字が小さいほど、例外 期間の優先順位が高くなります。例えば、「P06」の例外期間は、「P10」の 例外期間より優先されます。

タブページの下部では、例外期間の詳細を指定します。
 [日数]フィールドの値によって、次のように例外期間が定義されます。

[実行日]フィールドの値	例外期間
毎日	例外なくその年のすべての日が例外期間になります。

[実行日]フィールドの値	例外期間
日付指定	[日]、[月]、および[年]チェックボックスを使って定
	<u> </u>
第1	[月]および[年]チェックボックスをオンにした月および
	年の、【日】チェックボックスで定義した曜日が例外期間に
	なります。
	例
	毎月「第1」金曜日
	[月]および[年]チェックボックスをオンにした月および
	年の、[日]チェックボックスで定義した曜日が例外期間に
	なります。
	例
	9月の「第2」月曜日
最後から2番目	[月]および[年]チェックボックスをオンにした月および
	年の、【日】チェックボックスで定義した曜日が例外期間に
	なります。
	例
	11月の「最後から2番目」の水曜日
最後	[月]および[年]チェックボックスをオンにした月および
	年の、【日】チェックボックスで定義した曜日が例外期間に
	なります。
	例
	2000年の毎月「最終」火曜日

#### 例

Taltek社の従業員の休日は以下のように指定されています。

- 各月の第1金曜日は休日
- 8月中は、午前8時30分から午後12時30分までの午前中だけ勤務する。

#### 規則1:各月の第1金曜日は休日

- 1 [新規作成]をクリックします。
- 2 例外期間は、各月の第1金曜日です。この場合は、[月]および[年]チェックボックスをオフのままにして、例外期間が月や年に関係ないことを指定します。例外期間は金曜日だけなので、[日]チェックボックスはオンにします。
- 3 **[日数]**フィールドを「第1」に設定し、その隣のフィールドに「金曜日」 を指定します。

## 規則2:8月中は、午前8時30分から午後12時30分までの午前中だけ勤務する

- 1 [新規作成]をクリックします。
- 2 例外期間は、8月だけです。この場合は、[日]および[年]チェックボッ クスはオフにしたまま、[月]チェックボックスをオンにして、「8月」に 設定します。
- 3 例外期間は選択した日(8月いっぱい)だけになるので、[日数]フィール ドを「日付指定」に設定します。
- 4 この期間中は、従業員は午前中だけ勤務します。[作業時間]フィールドで 午前8時30分から午後12時30分までの時間を指定します。

## カレンダをチェックする

[プレビュー]タブページでは、[予定表]および[除外]タブページで定義 した規則を適用した結果を確認できます。確認する期間をテスト枠内の[開始 日]および[終了日]フィールドを使って指定します。

図 11.2. カレンダの詳細画面の [ プレビュー ] タブページ



- [開始日に関連するカレンダ]オプションには、デフォルトで、指定した [開始日]を含む週の業務時間のプレビューが表示されます。
- [経過時間]フィールドには、選択した期間中の業務時間の合計時間が表示 されます。

AssetCenter

💋 注意:

[開始日]を指定してから、[経過時間]フィールドに時間を入力すると自動的に[終了日]が計算されます。





本章では、タイムゾーン機能の使用方法について説明します。

#### 💋 注意:

この機能を使用するには、専用のAssetCenterライセンス契約を取得する必要が あります。

## タイムゾーンを管理する理由

クライアントマシンとデータベースサーバは地理的に別々の場所に配置できるので、AssetCenterではグリニッジ平均時(GMT)を基準にタイムゾーンと時差を管理します。次の規則が適用されます。

- クライアントマシンでは、「日付+時刻」型のフィールドはすべて、クライ アントマシンのタイムゾーンを反映して表示されます。
- サーバには、「日付+時刻」型のフィールドはすべて、定義されたタイム ゾーンを参照した値が格納されます。
- 日付と時刻に関するすべての計算には、時差も反映されます。

例

パリ(フランス)のタイムゾーンに従ってインデックス付けされたデータを持 つサーバがニューヨークに置かれ、2台のクライアントマシンがロンドンとパリ に配置されているとします。まず、グリニッジ平均時に従ってこれらの各クラ イアントマシンのタイムゾーンを定義します。

- サーバのタイムゾーン = GMT-5
- パリのクライアントのタイムゾーン = GMT+1
- ロンドンのクライアントのタイムゾーン=GMT
- データのタイムゾーン = GMT+1

すべての「日付+時刻」型の値は、サーバ上でGMT+1形式で格納され、パリの クライアントではGMT+1、ロンドンのクライアントではGMTとして表示されま す。例えば、ロンドンのクライアントマシンで作業指示を行う場合、作業指示 の解決期限が1998年5月15日の17時30分に設定されているとします。この場合、 他のマシンには次のように表示されます。

- サーバ:1998年5月15日午後12時30分
- パリのクライアント: 1998年5月15日午後6時30分
- ロンドンのクライアント: 1998年5月15日午後5時30分

## タイムゾーン機能を実装する

AssetCenterでタイムゾーンを正しく処理するには、次の操作を実行する必要が あります。

- AssetCenter Database Administratorでデータベースを作成する時に[タイムゾーンの使用]オプションを使って、タイムゾーンを定義します。
- 2 タイムゾーンを作成します(例えば、タイムゾーンに関連する情報をイン ポートします)。
- 3 [ツール/タイムゾーン]メニューを使って、マシンのタイムゾーンを定義 します。
- 4 タイムゾーンに従ってカレンダを定義します。

## タイムゾーンを作成する

AssetCenterのタイムゾーンの管理はWindowsと異なり、年間を通じて時間の変更 規則を考慮します。これにより、経過した現地時間をより正確に表示できます。 AssetCenterではタイムゾーン情報を使って、次の操作を行うことができます。

- 夏時間を反映したローカルの日付および時刻を表示します。
- 別の場所の時間を表示します。

タイムゾーンを手動で定義する手間を省くために、AssetCenterには、主要なタ イムゾーンを含む記述ファイルが添付されています。このファイルは、次の手 順でインポートできます。

- 1 **[ファイル/インポート]**メニューを選択します。インポートの選択ウィン ドウが開きます。
- 2 をクリックして[スクリプトの実行]を選択します。回をクリックして実行するスクリプトを選択します。選択するファイルは、AssetCenterのインストール先フォルダの/datakit/standardサブフォルダにある「tz.scr」です。
- 3 [インポート]をクリックします。スクリプトに従って、インポートが実行 されます。

## タイムゾーンを管理する

ここでは、**[夏時間]**(SQL名:memDaylightInfo)フィールドについて詳しく 説明します。

## 夏時間フィールドのフォーマット

[夏時間] (SQL名: memDaylightInfo) フィールドは次のような構造をとります。

<Year>=<DaylightInfo>|<Year> =<DaylightInfo>|<Year>=<DaylightInfo>|...

この節では、これらの用語を次のように表記します。

- <Year>=<DaylightInfo>をまとめて「パラメータ」と表記
- <Year>と<DaylightInfo>は、それぞれ「引数」と表記

<Year>および<DaylightInfo>引数に応じて変化する夏時間の概要は以下の表の通 りです。

	<daylightinfo>引数に値なし</daylightinfo>	<daylightinfo><b>引数に値があ</b> る</daylightinfo>
<year>引数に値なし (<year>=は表示されない)</year></year>	指定しているタイムゾーン 全体について夏時間は変更 されません。	<year>引数を持つパラメー タで定義された年を除くす べての年について夏時間情 報が有効になります。</year>

AssetCenter

	<daylightinfo><b>引数に値なし</b></daylightinfo>	<daylightinfo><b>引数に値があ</b> る</daylightinfo>
<year>引数に値がある</year>	適用できません。	<year>引数で指定した年か ら次の<year>引数までの各 年について、指定している タイムゾーンの夏時間情報 が有効になります。</year></year>

## <Year>引数の値

<DaylightInfo>で定義された夏時間変更情報を適用する年を指定する<Year>引数 には、4桁の年値を指定します(1990、1997、1998、2012など)。

## <DaylightInfo>引数の値

<DaylightInfo>引数の完全な形式は次の通りです。

<pre><stdshift>,<ditshift>,<sday></sday></ditshift></stdshift></pre>
, <smonth>,<sdaypos>,<shour></shour></sdaypos></smonth>
, <dday>,<dmonth>,<ddaypos>,<dhour></dhour></ddaypos></dmonth></dday>

この引数は、下に示す複数の副引数で構成されています。

副引数	説明	可能な値
<stdshift></stdshift>	タイムゾーン内の標準時間と対象と なるタイムゾーンの時間との時差を 分単位で示します。 例えば、パリ(GMT+1タイムゾー ン)の場合、 <stdshift>を30(分) に設定すると、このタイムゾーン内 の標準時間はGMT+1時間ではなく、 GMT+1時間30分となります。</stdshift>	デフォルトでは、この副引数はNULL ですが、任意の数値を設定できま す。ユーザは、この副引数の一貫性 を確認する必要があります。
<ditshift></ditshift>	夏時間と対象タイムゾーンの時間と の時差を分単位で示します。	デフォルトでは、この副引数は60に 設定されます(夏時間と参照時間 「GMT+」間の1時間の時差に対応 する)が、任意の数値を設定できま す。ユーザは、この副引数の一貫性 を確認する必要があります。
副引数	説明	可能な値
---------------------	------------------------------	-------------------------------------------------------
<sday></sday>	夏時間から標準時間に変更する日	"月曜日"
		"火曜日"
		"水曜日"
		"木曜日"
		"金曜日"
		"土曜日"
		"日曜日"
		値なし(この場合は、 <ddaypos>を 1から31の値に設定して、標準時間</ddaypos>
		から夏時間に変更りる日を指定りる 必要があります。)
<smonth></smonth>	夏時間から標準時間に変更する月	"1月"
		"2月"
		"3月"
		"4月"
		その他
		"11月"
		"12月"
<sdaypos></sdaypos>	夏時間から標準時間に変更する月内	"第1"
	の日の位置	"第2"
		"第3"
		"第4"
		"最後"
		"最後から2番目"
		<dday>に値がない場合は、1から31 の値</dday>
<shour></shour>	夏時間から標準時間に変更する時刻 (夏時間で指定)	24時間形式 ( HH:MM:SS ) の値

副引数	説明	可能な値
<dday></dday>	標準時間から夏時間に変更する日	"月曜日"
		"火曜日"
		"水曜日"
		"木曜日"
		"金曜日"
		"土曜日"
		"日曜日"
		値なし(この場合、 <sdaypos>を1か ら31の値に設定して、夏時間から標 準時間に変更する日を指定する必要 があります。</sdaypos>
<dmonth></dmonth>	標準時間から夏時間に変更する月	"1月"
		"2月"
		"3月"
		"4月"
		その他
		"11月"
		"12月"
<ddaypos></ddaypos>	夏時間から標準時間に変更する月内	"第1"
	の日の位置	"第2"
		"第3"
		"第4"
		"最後"
		"最後から2番目"
		<sday>に値がない場合は、1から31 の値</sday>
<dhour></dhour>	標準時間から夏時間に変更する時刻 (標準時間で指定)	24時間形式 ( HH:MM:SS ) の値

### 例

AssetCenter

「(GMT+01:00)パリ、マドリード、アムステルダム」タイムゾーンの夏時間 情報の例を見てみましょう。

2000=0,60,Sunday,October,Iast,03:00:00,Sunday,March,Iast,02:00:00| 0,60,Sunday,September,Iast,03:00:00,Sunday,March,Iast,02:00:00

使用されているパラメータは以下の通りです。

2000=0,60,Sunday,October,Iast,03:00:00,Sunday,March,Iast,02:00:00 0,60,Sunday,September,Iast,03:00:00,Sunday,March,Iast,02:00:00

## 最初のパラメータ



- <Year> = 2000は、次の<DaylightInfo>パラメータが2000年から後に適用されることを意味します。
- <StdShift>=0は、タイムゾーンとそのタイムゾーン内の標準時間に差がない ことを意味します。つまり、冬時間はそのタイムゾーンの時間と等しくなり ます。
- <DltShift>=60は、標準時間と夏時間の時差が60分、つまり1時間であること を意味します。夏時間は、タイムゾーンの時間に1時間を加えた値に等しく なります。
- <SDay> = Sundayは、標準時間への変更が日曜日に行われることを意味します。
- <SMonth> = Octoberは、標準時間への変更が10月中に行われることを意味します。
- <SDayPos> = Lastは、月内での日の位置を指定します。この例では、標準時間への変更は、10月の最後の日曜日に行われます。
- <SHour>=03:00:00は、標準時間への変更が午前3時に行われることを意味します。
- <DDay>=Sundayは、夏時間への変更が日曜日に行われることを意味します。
- <DMonth> = Marchは、夏時間への変更が3月に行われることを意味します。
- <DDayPos> = Lastは、月内での日の位置を指定します。この例では、夏時間への変更は、3月の最後の日曜日に行われます。
- <DHour>=02:00:00は、夏時間への変更が午前2時に行われることを意味します。

### 第2のパラメータ



- <Year>引数がないので、このパラメータは、前のパラメータで指定されてい ない年だけに有効であることを示します。
- <StdShift>=0は、タイムゾーンとそのタイムゾーン内の標準時間に差がない ことを意味します。つまり、冬時間はそのタイムゾーンの時間と等しくなり ます。
- <DltShift>=60は、標準時間と夏時間の時差が60分、つまり1時間であること を意味します。夏時間は、タイムゾーンの時間に1時間を加えた値に等しく なります。
- <SDay> = Sundayは、標準時間への変更が日曜日に行われることを意味します。
- <SMonth>=Septemberは、標準時間への変更が9月中に行われることを意味します。
- <SDayPos> = Lastは、月内での日の位置を指定します。この例では、標準時 間への変更は、9月の最後の日曜日に行われます。
- <SHour>=03:00:00は、標準時間への変更が午前3時に行われることを意味します。
- <DDay>=Sundayは、夏時間への変更が日曜日に行われることを意味します。
- <DMonth> = Marchは、夏時間への変更が3月に行われることを意味します。
- <DDayPos> = Lastは、月内での日の位置を指定します。この例では、夏時間への変更は、3月の最後の日曜日に行われます。
- <DHour>=02:00:00は、夏時間への変更が午前2時に行われることを意味します。

結果

#### 💋 注意:

2000年からは、標準時間への変更は、10月最後の日曜日の午前3時(時計を午前 2時に戻す)、夏時間への変更は、3月最後の日曜日の午前2時(時計を午前3時 に進める)に行います。

2000年より前の年は、標準時間への変更は9月最後の日曜日の午前3時、夏時間 への変更は3月最後の日曜日の午前2時に行います。

# AssetCenter Serverでタイムゾーンを管理する

AssetCenter Serverを使って、タイムゾーンのテスト方法を設定できます。[オプ ション/モジュールの設定]メニューを選択します。

# 実行するテスト

設定画面の**[全般]**タブページで、次のオプションからタイムゾーンのテストの実行方法を選択します。

- データベースサーバのタイムゾーンの検証
- サーバと比較したローカルマシンの時間の検証

これらの2つのテストはいずれも、データベースサーバの時刻とAssetCenter Server をインストールしたマシンの時刻を比較します。時間差は[(n\*30分)+m]で表さ れます。ただし、mは-15から+15分です。

### 2つのテストに共通する特徴

時間差が5分を超えると、AssetCenter Serverがインストールされているマシンの ローカル時刻を更新するメッセージが表示されます。

この更新を拒否すると(例えば、サーバの時刻を更新する場合)、接続が拒否 されます。データベースサーバの時刻とAssetCenter Serverをインストールしたマ シンの時刻のいずれかを変更して、2つの時刻の差を5分以内にすると、再接続 できるようになります。

## [データベースサーバのタイムゾーンの検証]オプショ ンの特徴

### 💋 注意:

この機能を正しく実行するには、AssetCenter Serverが稼動しているマシンで、時刻と夏時間の変更に関する情報を正しく設定しておく必要があります。

必要に応じて、AssetCenterのオプションテーブルで指定されているサーバのタ イムゾーンに関する情報が更新されます((n\*30分)の数値がサーバのタイムゾー ンに対応しない場合)。

### [サーバと比較したローカルマシン時間の検証]オプ ションの特徴

AssetCenterの内部操作に必要なサーバのタイムゾーンを取得します。

### テストの頻度

テストは次の場合に実行されます。

- 1 まず、AssetCenter Serverがデータベースに接続したときに実行されます。
- 2 その後は、AssetCenter Serverのモジュール設定画面([ツール/モジュール の設定])で定義するスケジュールに従って、定期的に実行されます。

### 操作に与える影響

タイムゾーンは、次のような様々な操作に影響を与えます。

- データベースの作成時
- データベースへの接続時
- インポートおよび出力時

# データベースの作成時

データベースの作成時にタイムゾーンオプションを定義できます。AssetCenter Database Administratorで[アクション/データベースの作成]メニューを選択し

<ul> <li>              SQLスクリプト/データペースの作成             データペース(B)             データペース(B)             データペース(C)             アーブルのテーブルスペース(D)             インデックスのテーブルスペース(D)             インデックスのテーブルスペース(D)             マテク・ブルスペース(D)             マースの作成(D)             マク・フレースの作成(D)             マースの作成(D)             マースの作成(D)             マース・フルドのへルブの使用(D)             インボート・ドするデーク             ・AutoCADDの統合・サンブルデータ             ・AutoCADDの統合・サンブルデータ             ・ログラアイル             マグフィルレ             マグ・ア・グレース             マーン             マーン</li></ul>	🛉 データベースの作成		
データペース(2)     データペース(2)     データペース(2)       インデックスのテーブルスペース(2)     インデックスのテーブルスペース(2)       インデックスのテーブルスペース(2)     インデックスのテーブルスペース(2)       「SOLスクリプトの作成(2)     ambase.sql       ● データペースの作成(2)     ●       ● データペースの作成(2)     ●       ● データペースの作成(2)     ●       ● インワリナトの作成(2)     ●       ● インワリナトの作成(2)     ●       ● インブーンの使用     ●       サーバのタイムゾーン     3MT+01:00       データのタイムゾーン     3MT+01:00       ● イムゴーンの使用     ●       サーバのヘルゴの使用(4)     ●       ● イムロ○AD統合の使用     ●       ● インボートドオるデータ     ●       ● AutoCADの統合 + サンブルデータ     ●       ● Automated Desktop Administration - 専門分野デ     ●       ● ログファイル     ●	素 SQLスクリプト/データペー	-スの作成	
データペース(2)         デーブルのテーブルスペース(2)         インデックスのテーブルスペース(2)         インデックスのテーブルスペース(2)         「シデータペースの作成(2)         ● データペースの作成(2)         ● データペースの作成(2)         ● ボータペースの作成(2)         ● ボータペースの作成(2)         ● ボータペースの作成(2)         ● ボータペースの作成(2)         ● 「オーダペースの作成(2)         ● 「オーダのクイムゾーン・(3MT+01:00         ● 「インボートジャナリオの使用(4)         ● 「インボート・ジャナリオの使用(4)         ● 「インボート・ジャナリオの使用(4)         ● 「インボート・ジャナリオの使用(4)         ● 「ムutomated Desktop Administration - 専門分野デ         ● 「日参ファイル」	J.		
デーダベース: WACDemo410ja テーブルのテーブルスペース① インデックスのテーブルスペース① 「ジデータペースの作成②) ● SOLスクリプトの作成③: ambase.sql ● 文字列の区切 BO ● データペースの作成②: ● 標準③ ● スクリプトの使用④: ambase.sql ● マクリの使用 サーバのタイムゾーン・ SMT+01:00 マイムゾーンの使用 サーバのタイムゾーン・ SMT+01:00 マイムゾーンの使用 ・ 「クオールドのヘルプの使用(④) 「AutoCADの統合・サンプルデータ ● AutoCADの統合・サンプルデータ ● AutoCADの統合・サンプルデータ ● AutoCADの統合・サンプルデータ ● AutoCADの統合・サンプルデータ ● AutoCADの統合・サンプルデータ ● AutoCADの統合・サンプルデータ ● AutoCADの統合・サンプルデータ	「データベース( <u>B</u> )		
テーブルのテーブルスペース①         インデックスのテーブルスペース①         ● データペースの作成②         ● データペースの作成②:         ● ボータペースの作成②:         ● ボータペースの作成③:         ● ボータペースの作成④:         ● ボータペースの作成④:         ● ボータペースの作成④:         ● オーバのタイムジーン:         SMT+01:00         ダームガラークの使用         サーバのタイムジーン:         「データのタイムジーン:         SMT+01:00         ダールドのヘルガの使用④         「ヘルドマヘルガの使用●         インボードするデータ:         ● AutoCADの統合・サンブルデータ         ● AutoCADの統合・サンブルデータ         ● Automated Desktop Administration - 専門分野デ         ● ログファイル	データベース: 🦉 (	AC De mo410ja	• 🖃
インデックスのテーブルスペースの         ● データペースの作成(Q)         ● SQLスクリプトの作成(Q):         ● データペースの作成(Q):         ● ボータペースの作成(Q):         ● ボータペースの作成(Q):         ● ボータペースの作成(Q):         ● ボータペースの作成(Q):         ● オーバのタイムゾーン:         ターバのタイムゾーン:         SMT+01:00         データのタイムゾーン:         SMT+01:00         ダンマールドのヘルブの使用(H):         ● AutoCADの統合の使用         インボードするデータ:         ● AutoCADの統合:         ● Tyブルデータ         ● AutoCADの統合:         ● Tyブルデータ         ● AutoCADの統合:         ● Tyブルデータ         ● AutoCADの統合:         ● Tyブルデータ         ● AutoCADの統合:	テーブルのテーブルスペース( <u>T</u> ):		
<ul> <li>✓ データベースの作成(2)</li> <li>○ SQLスクリプトの作成(2):</li> <li>◎ データベースの作成(2):</li> <li>◎ データベースの作成(2):</li> <li>◎ ボータベースの作成(2):</li> <li>◎ ボータベースの作成(2):</li> <li>◎ ボータベースの作成(2):</li> <li>◎ オムゾーンの使用</li> <li>サーバのタイムゾーン:</li> <li>○ SMT+01:00</li> <li>(4)</li> <li>○ マイールドのヘルブの使用(2):</li> <li>○ SMT+01:00</li> <li>(4)</li> <li>○ マイールドのヘルブの使用(2):</li> <li>○ AutoCAD総合の使用</li> <li>○ インボートジオリオの使用(2):</li> <li>○ AutoCAD総合の使用</li> <li>○ インボートするデータ:</li> <li>○ AutoCAD総合の使用</li> </ul>	インデックスのテーブルスペース Φ:		
○Sol スクリプトの作成(Q):       ambase.sql       文字列の区切(a0)         ●データベースの作成(Q):       ●         ● 標準(S)       ○スクリプトの使用(Q):         ○オンリプトの使用(Q):       ambase.sql         ● タイムゾーンの使用       ●         サーバのタイムゾーン:       GMT+01:00         データのタイムゾーン:       GMT+01:00         マイールドのヘルプの使用(H)       ●         ● AutoCAD統合の使用       ●         ● インボートシナリオの使用(M)       ●         インボートするデータ:       ● AutoCADの統合 - サンプルデータ         ● Automated Desktop Administration - 専門分野デ       ●         ● ログファイル       ●			
<ul> <li>●データペースの作成@):</li> <li>● 標準@)</li> <li>○ スクリプトの使用@):</li> <li>■mbase.sql</li> <li>● タイムゾーンの使用</li> <li>サーバのタイムゾーン:</li> <li>③MT+01:00</li> <li>ダークのタイムゾーン:</li> <li>③MT+01:00</li> <li>④ マコールドのヘルゴの使用(出)</li> <li>○ AutoCAD統合の使用</li> <li>● インボートシナリオの使用(出)</li> <li>○ インボートシナリオの使用(出)</li> <li>○ インボートライン「ク」</li> <li>○ AutoCADの統合 - サンブルデータ</li> <li>○ AutoCADの統合 - サンブルデータ</li> <li>○ AutoCADの統合 - サンブルデータ</li> <li>○ ログファイル</li> </ul>	OSQLスクリプトの作成( <u>G</u> ): amba	se.sql 文字列の区切GO	Ŧ
<ul> <li>● 標準⑤)</li> <li>○ スクリプトの使用(①): ambase.sql</li> <li>● タイムゾーンの使用</li> <li>サーバのタイムゾーン: 3MT+01:00</li> <li>データのタイムゾーン: 3MT+01:00</li> <li>● マイールドのヘルゴの使用(①)</li> <li>● AutoCAD統合の使用</li> <li>● インボートするデータ: AutoCADの統合 - サンブルデータ</li> <li>● Automated Desktop Administration - 専門分野デ</li> <li>● 同述 - サンブルデータ</li> </ul>	●データベースの作成( <u>C</u> ):	- 1 - 20 - 20	
○スクリブトの使用(型:         ambase.sql           ● タイムゾーンの使用         サーバのタイムゾーン:           「サーバのタイムゾーン:         GMT+01:00           データのタイムゾーン:         GMT+01:00           「マイールドのヘルブの使用(型)         AutoCADの統合の使用           「ハボートシナリオの使用(型)         AutoCADの統合 - サンブルデータ           「回達 - サンブルデータ         「回達 - サンブルデータ	●標準(S)		
<ul> <li>✓システムデータの作成①</li> <li>タイムゾーンの使用</li> <li>サーバのタイムゾーン: 3MT+01:00</li> <li>データのタイムゾーン: SMT+01:00</li> <li>SMT+01:00</li> <li>マフィールドのヘルブの使用(出)</li> <li>AutoCADの統合の使用</li> <li>インボートジナリオの使用(M)</li> <li>インボートするデータ: AutoCADの統合 - サンブルデータ</li> <li>Automated Desktop Administration - 専門分野デ</li> <li>□ 四道 - サンブルデータ</li> <li>ログファイル:</li> </ul>	○スクリプトの使用(U): amba	se.sql	
タイムゾーンの使用         サーバのタイムゾーン:SMT+01:00         データのタイムゾーン:SMT+01:00         マイールドのヘルブの使用(出)         AutoCAD統合の使用         インボートシナリオの使用(出)         インボートするデータ:         AutoCAD統合の使用         ログフィールドの         サンブルデータ         AutoCAD統合の使用	┌▼システムデータの作成(型) ────		
サーバのタイムゾーン: GMT+01:00 データのタイムゾーン: GMT+01:00 <b>○ フィールドのヘルブの使用(出)</b> 「AutoCAD統合の使用 インボートジオリオの使用(M) インボートするデータ: AutoCADの統合 - サンブルデータ Automated Desktop Administration - 専門分野デ ■明達 - サンブルデータ	「□タイムゾーンの使用		
データのタイムゾーン・ GMT+01:00 ▼フィールドのヘルブの使用(出) 「AutoCAD統合の使用 インボートジナリオの使用(M) インボートするデータ: AutoCADの統合 - サンブルデータ Automated Desktop Administration - 専門分野デ 可珍ファイル	サーバのタイムゾーン: GMT	+01:00	9
<ul> <li>マフィールドのヘルブの使用(出)</li> <li>AutoCAD統合の使用</li> <li>インボートシナリオの使用(M)</li> <li>インボートするデータ: AutoCADの統合 - サンブルデータ</li> <li>Automated Desktop Administration - 専門分野デ</li> <li>回達 - サンブルデータ</li> </ul>	データのタイムゾーン: GMT	+01:00	٩
■ AutoCAD統合の使用 インボートシナリオの使用(M) インボートするデータ、■ AutoCADの統合 - サンブルデータ ■ Automated Desktop Administration - 専門分野デ ■ 同達 - サンブルデータ ログファイル	▼ フィールドのヘルブの使用(出)		
インボートシナリオの使用(M) インボートするデータ AutoCADの統合 - サンブルデータ Automated Desktop Administration - 専門分野デ 回避達 - サンブルデータ ログファイル:	□AutoCAD統合の使用		
インボートするデータ AutoCADの統合 - サンブルデータ Automated Desktop Administration - 専門分野デ 同道 - サンブルデータ ログファイル	「□ インポートシナリオの使用( <u>M</u> ) ―――		
ログファイル·	インボートするデータ: 🗖 Au	itoCADの統合 - サンプルデータ	
ログファイル:		ite - サンプルデータ	-
	ログファイル:		
▼エラー発生時にインボートを中止(E)	▼⊥	ラー発生時にインポートを中止(E)	
作成(2) 閉じる(2)		作成(©) 閉じる	5( <u>C</u> )

ます。**[システムデータの作成]**セクションでタイムゾーンに関するオプションを指定できます。

[**タイムゾーンの使用**]オプションを使って、データベースの作成時にタイム ゾーン機能を使用するかどうかを指定します。

- チェックボックスをオンにすると、データベースの作成時にタイムゾーンが 使われます。
- チェックボックスをオンにしない場合は、データベースの作成時にタイム ゾーンは使用されません。

**[サーバのタイムゾーン]**と**[データのタイムゾーン]**では、データベースを 作成した時点で、サーバ上で有効にするタイムゾーンとデータを保存する時に 使うタイムゾーンを指定します。

#### 💋 注意:

このオプションは、データベースを作成するときだけ使用できます。サーバと データが参照する時間を定義できます。これらの値を変更しても、データベー ス内の既存の「日付+時間」値には影響しません。

# データベースへの接続時

AssetCenterは、データベースに接続する時に、「Am.ini」ファイル(Windowsの ルートディレクトリにある)からクライアントマシンのタイムゾーンを定義す る「LocalTimeZone」エントリを検索します。

この情報が見つからない場合は、AssetCenterはシステムのタイムゾーン(Windows NTまたはWindows 95で定義)を使います。

次に、AssetCenterは、「am.ini」の「LocalTimeZone」エントリに対応するタイ ムゾーンまたはWindowsで定義されたタイムゾーンを、データベース内で検索 します。

データベースへの接続時に「Am.ini」に加えられる変更は、以下の表の通りです。

「Am.ini」に	タイムゾーンのテーブルに	🕻 Am.ini
「LocalTimeZone」がある	対応するタイムゾーンがあ	(「LocalTimeZone」エン
かどうか	るかどうか	トリ)に保存される情報
可	可	データベースのタイムゾー
		ン
	いいえ	変更なし
いいえ	可	データベースのタイムゾー
		ン
	いいえ	システムのタイムゾーン

### クライアントマシンで時間を修正する

リモートデータベースに接続すると、AssetCenterはクライアントマシンのクロック時刻の有効性をサーバクロックと比較して検証します。

AssetCenterは、クライアントクロックのタイムゾーンを計算してから、2台のマシン間のクロック差を決定します。次の方法で計算します。

クロック差 = 剰余((サーバとローカルマシン間の分単位の時差)/30)

💋 注意:

剰余は、割算の余りです。

例えば、次のマシンがあるとします。

- マシンAのタイムゾーンはGMTで、18:02である。
- マシンBのタイムゾーンはGMT+1で、18:19である(つまりマシンAでは17:19 で、マシンAとは43分の差がある)。

クロックの差 = 母数 (43/30)= 13分

この差が5分(固定値)を超えると、AssetCenterはクライアントマシンのクロック時間を修正するメッセージを表示します。

ユーザがこの修正を拒否すると、接続は失敗します。

AssetCenterはこのチェックが定期的に実行し、クライアントマシンで時間が変更された場合もチェックします。デフォルトでは、60分置きにテストを実行しますが、次のように「Am.ini」(Windowsのルートフォルダにある)の[option] セクションにある「g\_ITimeZoneCheckInMns」オプションでこの頻度を変更することもできます。

[option]

g\_ITimeZoneCheckInMns = 30

上の例では、クロック差のチェックの頻度は30分に設定されています。

この頻度は、AssetCenterの**[データベースサーバのタイムゾーンの検証]**オプ ションでも設定できます。

#### 💋 注意:

この検証機能が使えるのは、タイムゾーン機能を有効にして作成したデータベー スだけです。

## インポートおよび出力時

この2つの機能の実行時は、データベース内のすべての「日付+時刻」フィール ドが、インポートまたは出力を行うマシンのタイムゾーンに対応しているもの としてデータが変換されます。





特殊フィールドの作成画面にアクセスするには、[管理/特殊フィールド]メ ニューを使います。

# 特殊フィールドの定義

特殊フィールドは、ユーザ定義の計算式を使って、他のフィールドや変数の値 に応じて値が計算されるフィールドです。3種類の特殊フィールドがあります。

- AQL
- Basic
- 特殊文字列

これらの特殊フィールドは、それぞれ異なる言語の計算式を使用しており、特定のオブジェクトにしかリンクできません。例えば、フィルタで使用できるのはAQLタイプの特殊フィールドだけです。

#### 💋 注意:

特殊フィールドは、読取り専用の仮想フィールド(データベースには計算式だけが格納される)です。必要な数の特殊フィールドを定義し、それぞれにユー ザ権限を割り当てることができます。

# 特殊フィールドの用途

特殊フィールドを使うと、追加情報を定義したり、AssetCenterデータベースの テーブルのレコードに関する合成情報を計算したりすることができます。次の いくつかの点を除き、特殊フィールドは通常のデータベースフィールドと同じ です。

- 通常のフィールドと異なり、特殊フィールドの値は、AssetCenterデータベー スには格納されません。
- 特殊フィールドの値は、ユーザではなく計算式によって入力されます。
- 特殊フィールドを特定のフィールドの1個のレコードに関連付けることはできません。データベース内の他の通常のフィールドと同様、特殊フィールドは、テーブルのすべてのレコードに関連付けられ、そのテーブルの各レコードに対して値(NULLも可)を持ちます。
- 特殊フィールドはレコードの詳細画面には表示されません。リストにのみ表示できます。
- フィールドのタイプが[特殊文字列]または[BASICスクリプト]の場合は、特殊フィールドは標準フィールドのデフォルト値の計算だけに使用できます。

# 特殊フィールドを作成する

特殊フィールドを作成する前に、各特殊フィールドの特徴を理解することが大 切です。

ここでは、次の内容について説明します。

- 特殊フィールドの概要
- 特殊フィールドの作成方法

### 特殊フィールドの概要

特殊フィールドのタイプはそれぞれ異なるプロパティを持ち、それによって使 用方法も異なります。

3つのタイプの主な相違点は次の表の通りです。

表 13.1. 特殊フィールドのタイプ

フィール ドのタイ プ	フィール フィールドの特性 ドのタイ プ			計算を実 行するコ ンピュー タ	計算式で使われる言語 の特徴	
	表示可能	並べ替え 可能	フィルタ で使用可 能		利点	欠点
AQL	可	可	可	データ ベース サーバ	強力 統合エ ディタ	限言 こプフドフ値用せ に語 のの ーデルは オにで た して し た し た し た の の ー デ ト 使 ま の の ー デ い し た に ろ の の ー に う に で い の の の ー に う に の の の っ に が の の の っ に が り の の っ に ろ の の っ に で の し の う に ろ し の の っ に ろ し の う と う と う ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ ろ
特殊文字列	可	可	いいえ	クライアント	単純	強な字フドびの純結 フドリーお字の連 (と) (と) (と) (と) (と) () () () () () () () () () () () () ()
Basic	可	いいえ	いいえ	クライア ント	多くの可 能性 柔軟	このタイ プの フィール ドは表示 のみが可 能です。

#### 💋 注意:

この表から、AQLタイプのフィールドは他の2つの特殊フィールドに比べ、非常に用途が広いことが分かります。

AQLクエリでは、3つのプロパティ(表示可能、並べ替え可能、フィルタで使用可能)をすべて活用できます。

特性	対応するAQL <b>引数</b>
表示可能	SELECT 句
並べ替え可能	SELECT
	ORDER BY
	GROUP BY 句
フィルタで使用可能	SELECT
	ORDER BY
	GROUP BY
	WHERE
	HAVING 句

AQLクエリの詳細については、『AssetCenter**の高度な使い方』**、「AQL**クエリ」**の章を参照してください。

### サーバ / クライアントマシンによる計算

AQLタイプのフィールドの場合、必要な計算はデータベースサーバで実行され、 結果がクライアントマシンに返されます。このため、クライアントマシンの処 理速度への影響はまったくなく、ネットワークトラフィックが軽減されます。 これに対し、データベースに送信されるSQLクエリはより複雑です。

## 特殊フィールドの作成方法

ここでは、特殊フィールドの作成方法について詳しく説明します。

### 用途を分析する

次の2つの基準に従って、フィールドのタイプを選択します。

- フィールドのタイプのプロパティ。表示、並べ替え、フィルタまたはデフォ ルト値として使用できるかどうか。
- 使用するタイプに応じてかかる「コスト」。使用状況と計算式の複雑さを比較します。3種類の特殊フィールドをコストの高い方から見ると次のようになります。
  - 特殊文字列
  - AQL

AssetCenterの高度な使い方

Basic

#### 💋 注意:

可能な限り、最も「経済的」なタイプのフィールドを使うようにしてください。

例

- 情報のみの場合は、Basicタイプの特殊フィールドで十分です。
- フィールド値に基づいてレコードを並べ替える場合は、AQLまたは特殊文字 列を使う必要があります。
- フィールドの値に応じてフィルタを使ってレコードを検索する必要がある場合は、AQLを使う必要があります。

用途が明確になったら、次の手順に進みます。

### 作成画面を開く

[管理/特殊フィールド]メニューを選択します。特殊フィールドの作成画面 が表示されます。

タイトル:	年間TCO				
SQL名: cf_AvgYearlyTco					
説明:	年間TCO				
テーブル:	■資産 (am Asset) ・				
フィールドタイプ:	BASICスクリプト・				
結果のデータ型: 数値 ▼					
-BASICシンタックス					
Const DAYS_PER_YEAR = 365.25 'Def. life expectancy = 3 years Const DEFAULT_LIFE_EXPECTANCY					
Dim IRc As Long					
•					

### 特殊フィールドの情報を入力する

まず、この画面の上部に入力して、特殊フィールドに固有の情報を指定します。

- **[タイトル]**(SQL名: Label)フィールドには、リストの列ヘッダーとして 使う、特殊フィールドのラベルを入力します。
- [SQL名](SQL名:SQLName)フィールドには、特殊フィールドのSQL名 を入力します。この名前には、「cf\_」という接頭語が付き、例えば、Basic スクリプト、クエリ、またはフィルタでこのフィールドを参照する場合に使 います

#### 💋 注意:

一度作成したフィールドのSQL名は変更できません。SQL名を変更すると、 そのフィールドへの参照は無効になります

 [説明](SQL名: Description)フィールドには、フィールドを表示するリ スト(フィルタ、リストの設定画面など)で使う、フィールドの短い説明を 入力します。

## フィールドを使うコンテキストを定義する

[テーブル] (SQL名: TableName) および[フィールドタイプ] (SQL名: seType) フィールドでは、特殊フィールドを使うコンテキストを定義できます。

- [テーブル](SQL名: TableName)で、特殊フィールドを関連付けるテーブルを指定できます。フィールドはこのフィールドで指定したテーブルでのみ使用できます。
- [フィールドタイプ](SQL名: seType)フィールドで、特殊フィールドの タイプを指定できます。このタイプによって、フィールドのプロパティ(表 示、並べ替え、またはフィルタが使用可能どうか)が変わります。
- [結果のデータ型]フィールドで、特殊フィールドの結果タイプを指定でき ます。このタイプによって、表示形式が変わります。結果タイプが日付型の 特殊フィールドは、データベース内の他の「日付」型のフィールドと同じよ うに表示されます。

### フィールドの計算式を入力する

フィールドの計算式を入力します。画面下部のテキストフィールドに直接入力 するか(表示されるフィールドは、選択する属性タイプによって変わります)、 図をクリックし、[F4]キーを押して、対応するエディタ画面にアクセスしま す.

#### 💋 注意:

使用する言語は、フィールドのタイプによって異なります。

計算式の作成に使用できる言語の詳細については、次のマニュアルを参照して ください。

- Basic言語の場合は「AssetCenterの高度な使い方」、「スクリプト」の章。使用する関数はRetVal()です。
- AQL言語の場合は、「AssetCenterの高度な使い方」、「AQLクエリ」の章。
- 特殊文字列の場合は、『管理』、「データベースの標準記述ファイル」の 章、「テーブルについて」の節、「テーブルの表記名」。

### 特殊フィールドのユーザ権限を定義する

[管理/ユーザ権限]メニューを選択します。ユーザ権限を作成する画面が表 示されます。

#### 💋 注意:

特殊フィールドには、読取専用でのみアクセスできます。

- [説明](SQL名: Description)フィールドにユーザ権限の短い説明を入力し、必要に応じて、[コメント](SQL名: Comment)フィールドにコメントを入力します。
- 2 特殊フィールドに関連するテーブルのツリー構造を展開します。 ☆ (特殊フィールド) アイコンが付いたブランチに、そのテーブルの特殊フィールドの完全なリストが含まれます。
- 3 ユーザ権限を編集するフィールドを選択します。[フィールド、リンク、任 意管理項目]枠の[読み取り]チェックボックスを使って、そのフィールド の読み取り権限を定義できます。このチェックボックスをオンにすると、こ のユーザ権限を持つプロファイルのユーザだけが、特殊フィールドを表示で きます。このチェックボックスをオフにすると、すべてのユーザにこのフィー ルドに対する(読み取り専用)権限が与えられます。

# 特殊フィールドの使用方法

特殊フィールドの使用方法はタイプによって異なります。タイプが使用目的に 合っていることを確認する必要があります。フィールドを表示するリスト(フィ ルタの作成、リストの設定など)では、ユーザに分かりやすいよう、使用可能 なフィールドのみが表示されます。

## リストの設定で特殊フィールドを使う

ポップアップメニューの**[リストの設定]**コマンドを使って、テーブルのすべてのレコード用の特殊フィールドの値を表示できます。 テーブルのフィールドおよびリンクのリストで、「特殊フィールド」のマークがついたプランチを開きます。このプランチには、使用可能な特殊フィールドのリストが表示されます。**[編集/オプション]**メニュー/**[表示]**項目で指

定したフォーマットで、特殊フィールドが表示されます。 他のフィールドと同じように、リストにこれらの特殊フィールドを追加できま す。

# テーブルのレコードをフィルタを使って検索する

テーブルのレコードをフィルタを使って検索するAssetCenterでは、AQLタイプ の特殊フィールドの値に応じて、テーブルのレコードを検索できます。これを 行うには、ポップアップメニューから[シンプルフィルタ]コマンドを選択し、 「特殊フィールド」ブランチに移動します。AQLタイプのフィールドだけが表 示されます。

# 特殊フィールドを参照する

「cf\_」という接頭語を持つSQL名を使って、特殊フィールドを参照できます。 下の画面は、特殊フィールドのSQL名をフィルタに使って、レコードを検索し た例です。

🤨 ボートフォリオ品目 '1 Compaq Deskpro 40005 (DKT0010)	で6:001076)'の詳細			_ 🗆 🗡
🍾 cf_self 🔹 🗞 △モデル	<u> </u>	数量 △使用単位(モデル)	) +	新規作成( <u>N</u> )
= 2 ↓ C ■ Deskprc 4000S		1		複製( <u>U</u> )
Fast Etherlink XI10/	100 PCI TX	1		削除( <u>D</u> )
Wirdows NT Workstat	ion 4.0	1	+	
Lardesk Virus Protec	+ 05.0	_⊗_	152 / 152	
🛄 🗆 🖂 🖂 🖂 🖂	076)		<b>へ</b> = -	分割( <u>D</u> )
モデル: Deskpro 4000S	۰ 🗉 ک	Code: 001076		タグ( <u>G</u> )
カテゴリ・/Π/コンピュータ/アスクトップ/				在庫( <u>0</u> )
全般 コンボーネント 調整 コンビュータ 履歴 :	書類			
割当:使用中	数量:	1		
使用開始日: 2002/03/05	ユーザ:	川崎 赤子	۲ 🗉 ک	
棚卸日:	責任者:		< ■ ▼	
Ret P P . A	設置場所:	/札幌事務所/ビル02/2階/	• 🗉 🔻	
	コストカテゴリ:		≪ ≣ ▼	
官連案件「四有資圧のジ	コストセンタ:	Π	< ≣ -	
単価: US\$622.69 🔷 🛢	現レコード:		< ≣ -	
			<u> </u>	
			_	
				- 第じる( <u>C</u> )

さらに、特殊フィールドは、AssetCenterの次のモジュールまたは機能でも使用 できます。

- AssetCenter Web
- AssetCenter API
- レポート
- 書式

AssetCenterの高度な使い方

これらのモジュールや機能については、対応するマニュアルを参照してください。



AssetCenterでは、独自のウィザードを作成したり、既存のウィザードを用途に 合わせて変更したりすることができます。ウィザードは、テキストフィールド として保存されます(**[ウィザードのスクリプト]**(SQL名:WizardScript) フィールド、**[ウィザード]**タイプアクションの詳細画面の**[ウィザード]**タ ブページ)。ウィザードを作成するには、このフィールドに直接コードを入力 するか、グラフィカルエディタを使います。これには、ウィザードの構造とこ の構造を記述するスクリプト言語をよく理解しておく必要があります。

# 表記法

ウィザードの構造の説明では、次の表記を使います。

#### 表 14.1. 表記法

[] 大括弧は、データベース内のフィールドの値(コンテキストを指定 するウィザードの場合)または「専用フィールド」のCurrentSelection およびCurrentTableのいずれかを参照する時に使います。大括弧はま た、オプションパラメータを示す際にも使用されます。

-		-	
Acc	ot	$\sim$	nto
855	1811	60	пе

<>	山形括弧は、プロパティの短い説明を示します。実際にプロパティ を入力するときは、山形括弧を使わずに、括弧内にあるテキストに 該当する情報だけを入力してください。
	縦線(パイプ)文字は、プロパティの値を区切るのに使います。ま た、複数の列からなるリストのタイトルと値を区切るときにも使い ます。
{}	中括弧は、ノードの定義または複数行にわたるプロパティのスクリ プトを囲むのに使います。また、ウィザードのプロパティの値を参 照する時にも使います。
1	Basicコードの例のアポストロフィは、ウィザードの実行時に AssetCenterが無視するコメント行を示します。
;または //	ウィザード内のセミコロンまたは二重スラッシュは、AssetCenterが 無視するコメント行を示します。

## 定義

ウィザードの構造の説明で使われている用語の定義を以下に示します。

### Twip

Twipは、ウィザードで使われるサイズの単位で、デフォルトでは長さを表します。画面の解像度とは関係ありません。次の単位に相当します。

- 1440twipは1インチに相当します。
- 567twipは1cmに相当します。
- 96dpi解像度(Windowsの標準解像度)では、15twipは1ピクセルに相当します。

## コントロール

コントロールは、データを編集する時に使うグラフィック要素です。一般的な コントロールには、チェックボックス、テキストエディタ、ボタン、ドロップ ダウンリストなどがあります。

#### ノード

ノードは、ウィザードのツリー構造の階層レベルに対応します。例えば、ノード「N」のサブノードは、ツリー構造内でノード「N」にリンクしている1レベル下のノードです。

AssetCenterの高度な使い方

#### 💋 注意:

ノード名には英数字のみを使用できます。ノード名の最大長は、22文字(半角の場合)です。

# オブジェクト

オブジェクトは、次の要素を示す総称的な用語です。

- ウィザード全体
- ウィザードのページ
- ページ内のコントロール(チェックボックス、テキストエディタ、ボタン、 ドロップダウンリストなど)
- 変数
- その他

# オブジェクトと子オブジェクト

オブジェクト「A」にオブジェクト「B」が含まれる場合

- オブジェクト「A」は、オブジェクト「B」の親オブジェクトになります。
- オブジェクト「B」は、オブジェクト「A」の「子オブジェクト」になります。

🌻 警告:

この概念は構成上の関係だけで、データの継承とは関係ありません。

# オブジェクトのフルネーム

オブジェクトのフルネームは、全ての親オブジェクト名とこのオブジェクトの 名前で構成されます。各オブジェクトは、ピリオド(.)で区切ります。



この例では、オブジェクト「C」のフルネームは、次のようになります。

<オブジェクトAの名前>.<オブジェクトBの名前>.<オブジェクトCの名前>

# 変数

変数は、ウィザードの実行中に変更可能なデータを格納する名前付きの保存領 域です。各変数には、ウィザード内で個別に識別される名前があります。 AssetCenterのウィザードには2種類の変数があります。

- LONG(倍長整数)またはSTRING(文字列)型のノードで定義するウィザー ド変数。ノードのタイプによって変数のタイプが決まります。LONGノード で定義する変数は倍長整数、STRINGノードで定義する変数は文字列です。 これらの変数は、定義上グローバルです。つまり、ウィザードのどのノード からも、これらのフルネームを使って参照できます。必要に応じて、これらの変数をAssetCenterで自動的に再計算することができます。
- ウィザード内のBasicスクリプトで使われるBasic変数。デフォルトでは、これらの変数はローカルですが、COMMONおよびGLOBALプロパティを使ってグローバルにすることができます。これらの変数は、AssetCenterによって自動的に再計算されることはありません。

# トランジション

トランジションとは、ウィザードのあるページから別のページに移ることを示 します。特定のページでいくつかのトランジションを定義できます。それぞれ のトランジションには、トランジションを有効にしてトリガするのに必要なユー ザ定義の条件があります。

- 例えば、ユーザが[次へ]をクリックすると、最初に有効なトランジション (条件が満たされたトランジション)が実行されます。有効なトランジションがない場合は、[次へ]ボタンは無効になります。
- ウィザードに必須プロパティがある場合、この必須プロパティが入力されていないと、[次へ]ボタンは使用不可能になります。
- ウィザードの全段階を完了せずに[終了]をクリックすると、指定されていない部分はデフォルト値になります。

# 構造モデル

AssetCenterウィザードは、ウィザードを実行するタイプのウィザードです。ウィ ザードは、構造スクリプトで定義します。

ウィザードスクリプト(ウィザード)はノードで構成されます。

- ウィザードの各ノードには、名前、1つまたは複数のサブノード、およびプロパティブロックがあります。ノードのタイプは次の通りです。
  - ROOTノード(起点ノード):このノードは固有で、他のすべてのサブ ノードを含みます。
  - STARTノード:このノードは固有で、ウィザードの起動時に実行される スクリプトが含まれます。
  - PAGEノード:ウィザードのページ情報を記述します。
  - TRANSTIONノード:2つの「PAGE」タイプのノード間のトランジション を記述します。
  - FINISHノード:このノードは固有で、ウィザードの最後に実行されるスクリプトが含まれます。
  - PARAMSノード:このノードは固有で、別のウィザードに渡すパラメータが含まれます。複数のウィザードを連続して(パラメータを受け渡す、受け渡さないに関係なく)実行できます。このようなウィザードのことを、連鎖しているといいます。
  - LONGノードまたはSTRINGノード:対応する変数のデータ型を定義します
- プロパティの値は定数またはBasicスクリプトを用いて明確に指定します。 Basicスクリプトの場合、値はスクリプト評価の結果です。

ウィザードは、トランジションでリンクされたページで構成されています。1つのページから別のページへの分岐は、ユーザが入力する選択情報によって決まります。ウィザードの構造は次の図の通りです。

#### 図 14.1. ウィザードの構造



# ウィザードページのモデル

ウィザードページは、次のように構成されています。

・ ウィザード: '移動'		•		_	
移動する	ユーザの選	択			
	移動するユーザ ▲部署名/姓 ● Taltek ● ファイナンス + 安西 + 山田 + 総務部 + 山田 ● 2戸 # 生産課 + 名庫 ● 工ま # 生産課 + 名庫 ● 田達員 + 木村 * 営理 * 研究開発課 * 情報システム部	▲ 名前 博子 功 致子	<ul> <li>/ 航在地</li> <li>/ 札幌事務所/</li> <li>/ 横浜ビル/31階/</li> <li>/ 横浜ビル/31階/024 - 事</li> <li>/ 横浜ビル/31階/025 - 事</li> <li>/ 札幌事務所/ビル02/含庫</li> <li>/ 札幌事務所/ビル02/含庫</li> <li>/ 札幌事務所/ビル02/含庫</li> <li>/ 札幌事務所/ビル02/含庫</li> <li>/ 札幌事務所/ビル02/含庫</li> <li>/ 札幌事務所/ビル02/含庫</li> <li>/ 札幌事務所/ビル02/21</li> <li>/ 札幌事務所/ビル02/21</li> <li>/ 札幌事務所/ビル02/21</li> <li>/ 札幌事務所/ビル02/21</li> <li>/ 札幌事務所/ビル02/21</li> <li>/ 札幌事務所/ビル01/21</li> </ul>	務所/ 務所/ // /002 - ホール/ /003 - ホール/ /001 - ホール/ /004 - 事務所/ /	+
	< 戻る(円)	次	<ul><li>へ№&gt;</li><li>終了(E)</li></ul>	キャンセル	٧

# ウィザードに関する一般情報

ウィザードのコード(ウィザードを使うアクションの詳細画面の[ウィザード] タブページの、[amAction]テーブルの[ウィザードのスクリプト](SQL名: WizardScript)フィールド)は、中括弧({})で区切られたブロックからなる構 造化テキストです。このテキストで、ウィザードの構造を定義します。構造は ツリー構造です。

ウィザードのツリーの各ノード(「ROOT」、「PAGE」など)には、サブノー ドとプロパティのセットを無制限にリンクできます。

# 一般的な構造とシンタックス

ノードの構造とシンタックスは次の通りです。

; これはスクリプト外のコメントです { <ノードタイプ> <ノード名>

AssetCenterの高度な使い方

```
<プロパティ名> = <プロパティの値>
'これはスクリプト内のコメント行です。
...
{ <プロパティ名> =
...
}
{ <ノードタイプ> <ノード名>
<プロパティ名> = <プロパティの値>
...
{ <プロパティ名> =
...
}
```

ノードは、次の規則に従って指定します。

- ノードの名前は付けなくてもかまいません。ノードに名前を付けない場合
   は、ノードに自動的に名前と番号が割り当てられます。
- ノード名にスペースを使うことはできません。
- ノードの名前が「=」の場合、これはノードではなく複数行プロパティとなります。複数行プロパティの詳細については、本章の「プロパティのシンタックス」を参照してください。
- スクリプト外のセミコロン(;)で始まる行と、スクリプト内のアポストロフィ(')で始まる行はコメントとして解釈され、無視されます。

💋 注意:

注意:中括弧({)とノードタイプ間にはスペースを入れる必要があります。スペースがない場合、ウィザードは実行されません。

# ノードのプロパティ

プロパティの値は、定数またはスクリプトを使って定義できます。定数は、数 値、ブール値、またはテキストのいずれかです。

#### 💋 注意:

オブジェクトに関連するプロパティは、任意にすることも必須にすることもで きます。プロパティは、「論理的」(オブジェクトの定義を補足)または「物 理的」(オブジェクトの視覚面に影響)に定義できます。

# 宣言的モデル

プロパティは、循環参照(A={B}、B={A})を定義する宣言モードに従って定 義します。

<プロパティ名>=<スクリプト>

この定義には従属関係のリストが関連付けられます。

A={B}+{C}

この例では、プロパティ「A」はプロパティ「B」および「C」に依存します。 つまり「A」に従属するリストは、「B」と「C」になります。

- この結果、プロパティは次の場合に変化します。
- この従属関係のリスト内のプロパティの1つが変化した場合。
- プロパティまたは従属プロパティをユーザが変更した場合。

### 定数をプロパティの値として定義する

次のシンタックスでは、プロパティの定数値を定義します。

- テキスト型のプロパティ
  - <プロパティ名>=<テキスト>
- ブール型のプロパティ
  - <プロパティ名>=TRUE
  - <プロパティ名> = FALSE
  - <プロパティ名> (<プロパティ名> = TRUEに相当)
- 数値型のプロパティ
  - <プロパティ名>=42
- <プロパティ名> = {<Basic変数またはプロパティのフルネーム>}

💋 注意:

ブール値「TRUE」は、「0」以外の数値と同じです。「FALSE」は数値「0」と 同じです

## プロパティを参照する

オブジェクトのプロパティを参照する(プロパティまたはオブジェクトの内容、 特に値を参照する)には、次のシンタックスを使います。

{<プロパティのフルネーム>}

AssetCenterの高度な使い方

ページ「Page1」のプロパティ「Prop」を参照する場合は、次のように入力します。

{Page1.Prop}

このシンタックスでは、大文字と小文字は区別されません。

# スクリプトをプロパティの値として定義する

# スクリプトの概念

スクリプトは、グローバル変数RetVal内の値を返す1行または複数行のBasicプロ グラムです。1行スクリプトの場合、変数は暗黙的です。複数行スクリプトの場 合は指定する必要があります。

すべてのBasicスクリプトについて、戻り値のデータ型に注意してください。こ のデータ型は、スクリプトを介して計算されるプロパティのタイプに依存しま す。

# 1行スクリプトのシンタックス

<プロパティ名>=<スクリプト>

例

Variable="The name is: " & {Name}

上の1行スクリプトは、次の複数行スクリプトと同じです。

```
{ Variable =
RetVal="Le nom est : " & {Name}
```

# 複数行プロパティのシンタックス

```
{ <プロパティ名 >=
|<スクリプト>
い
```

例

{ LABEL = IF {Page1.Title}="Chose an employee" THEN RetVal="Employee" ELSE RetVal="Department"

AssetCenter

END IF

## プロパティで適用できるメソッド

メソッドでは、プロパティまたはノードにリンクしている値を取得したり、こ のプロパティで関数を実行したりすることができます。この点で、メソッドは 高度な関数と考えることができます。

メソッドのシンタックスは次の通りです。

{ノード.ノード.ノード[.プロパティ][.メソッド([arg1[, arg2[...]]])]}

以下の値を入力します。

- ノード:ノード名
- プロパティ:プロパティ名
- メソッド:メソッド名
- arg1、arg2、…: 定数またはBasic式(Basic式で中括弧{}を使用してはなりません)。

💋 注意:

この例では、大括弧([])文字を使ってオプション項目を区切っています。

例えば、ページ「PAGE1」内の「LISTBOX」コントロールから行数を取得する には、このタイプのコントロールに関連するCOUNTメソッドを使います。コマ ンドは次の通りです。

{PAGE1.LISTBOX.VALUES.COUNT()}

# テーブルタイプのプロパティ

テーブルタイプのプロパティは、次の形式に従って値が定義されるプロパティ です。

<列  列  >=<行の識別子>, <列  列  列 >=<行の識別子>,	
これらのプロパティの値は、次のように表形式で表示できます。	

		<b>列</b> 1	<b>列</b> 2	<b>列</b> 3
<b>行番号:</b> 1	行の識別子	セル ( 1,1 )	セル ( 2,1 )	セル (3,1)
	( <b>191]:</b> 18)			

AssetCenterの高度な使い方

		<b>列</b> 1	<b>列</b> 2	<b>列</b> 3
<b>行番号:</b> 2	行の識別子 (例:29)	セル(1,2)	セル(2,2)	セル ( 3,2 )
<b>行番号:</b> 3	行の識別子 (例:78)	セル(1,3)	セル(2,3)	セル ( 3,3 )

#### 💋 注意:

識別子は「テキスト」タイプです。

### 例

[部署と従業員]テーブルのクエリ結果を値として含むLISTBOXノードの VALUESというプロパティがあるとします。このクエリはこのテーブルの各レ コード用に、[部署名/姓](SQL名:Name)および[名前](SQL名: FirstName)フィールドの値を返します。このプロパティには次の値が入ってい るとします。

VALUES="Colombo|Gerard=32,Lubeck|Alexander=64,Daquin|William=24" この値を、次の表形式で表示できます。

		名前	名前
1	32	Colombo	Gerard
2	64	Lubeck	Alexander
3	24	Daquin	William

# グローバル変数CurrentTableとCurrentSelectionを使う

これらの変数の内容は、次のシンタックスを使って取得できます。

[CurrentTable]		
[CurrentSelection]		

これら2つの変数の特徴は次の表の通りです。

変数名	変数の説明	コメント
CurrentTable	ウィザードの起動時にアク ティブなテーブルのSQL名 を含みます。アクティブな テーブルがない場合は空の 文字列を含みます。 文字列型の変数	この変数は、AssetCenterに より自動的に入力されま す。ユーザが値を手動で入 力することはできません。
CurrentSelection	ウィザードの起動時に選択 されたレコードの内部識別 子の、カンマ区切りのリス トを含みます。 文字列型の変数	この変数は、AssetCenterに より自動的に入力されま す。選択項目がない場合ま たはテーブルが指定されて いない場合、変数は空の文 字列を含みます。ユーザが 値を手動で入力することは できません。

# ウィザードの連鎖化

実行後のウィザードは、別のウィザードの実行をトリガしたり、このウィザードにパラメータ(変数)を渡したりすることができます。このようなウィザードを連鎖ウィザードと呼びます。

## 実行

AssetCenter

ウィザードAからウィザードBをトリガするには、FINISHノードにCHAINプロパ ティが必要です。このプロパティには、実行する**[ウィザード]**タイプアクショ ンのSQL名の値(この場合は「B」)を設定する必要があります。

# 作成パラメータ

パラメータは、ウィザードAのPARAMSノードを使って、ウィザードBに渡され ます。これらのプロパティは、ウィザードBのPARAMSノードのパラメータに追 加されます。ウィザードAのPARAMSノードとウィザードBのPARAMSノードの 両方で同じパラメータが定義されている場合は、ウィザードAのパラメータが優 先されます。

# Basic**関数**

AssetCenterの一般的な関数(AmCounter関数を除く)に加え、ウィザードでは次の関数を使用できます。

- AmComputeString()
- AmDecrementLogLevel()
- AmExecTransition()
- AmLog()
- AmMsgBox()
- AmPagePath()
- AmProgress()
- AmRefreshProperties
- AmSetProperty
- AmUpdateDetail
- AmValueOf
- AmWizChain

#### 🌻 警告:

ウィザードスクリプトからBasic関数を呼び出す場合は、関数が返す値を常に変数に割り当てる必要があります。変数に割り当てないと、エラーが返されます。 次の例はコンパイルされません。

AmGetFieldLongValue(hRecord, "IUserId", {IEmplDeptId})

正しいスクリプトは次の通りです。

Dim IValue as Long IValue=AmGetFieldLongValue(hRecord, "IUserId", {IEmpIDeptId})

# ROOT ノードの定義

ROOTノードでは、ウィザード全体を定義します。このノードは、ウィザード全体に適用できる一般的なプロパティブロックと、ウィザードに含まれるオブジェクトを表す一連のサブノードで構成されます。

# ROOT ノードのシンタックス

ROOTノードのシンタックスは次の通りです。

' ROOTノードの一般的なプロパティブロック NAME= IMAGE=
 ' ROOTノードのサブノードの定義 { FINISH
 }
{ PAGE 
{ TRANSITION
}

# ROOT**ノードのプロパティ**

ROOTノードで定義できるすべての論理 / 物理プロパティの一覧は、次の表の通 りです。

表 14.2. ROOT ノードの論理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
NAME="<ウィザード名 >"	ウィザードの名前を定 義します。 文字列型のプロパティ	NAME="Move Wizard"	このプロパティに値を 定義す。ウィザード名は22 文字の内(半角の場 合)で指定す。こ のプロパティは、ウィ ザードを連ます。この ウィザードにするた めにせードに関連する データは、ます。そのた め、名前をす。そのた お奨めします。
TITLE="<ウィンドウの タイトル>"	ウィザードのウィンドウのタイトルを定義し	TTTLE="Move Wizard"	このプロパティにはで きるだけ値を定義する
	ます。	VIZUIU	ようにしてください。
	文字列型のプロパティ		
GLOBAL=<スクリプト	Basicスクリプト (また	{GLOBAL=Dim	
>	はコード)が含まれま	Filter As	
	9。このノロハテイ け グローバル恋数	String}	
	は、ノロ ハル复数 (すべてのウィザード		
	に適用可能)および		
	ウィザードのすべての		
	ノードで使用できるグ		
	ローバル関数などを定		
	我9る時に役立らます。		
	このスクリプトは宣言		
	スクリプトのため、実		
	行されません。起動時 にフクリプトを宝行す		
	こへノリノーで天119 るには、STARTノード		
	を使用します。		
	文字列型のプロパティ		

プロパティ名=値	プロパティの説明	例	コメント
COMMON	全ウィザード内に自動 的に入るBasic関数が含 まれます。 このプロパティは、 データベースの作成時 にデータベースに挿入 される「gbbase.wiz」 ファイル(読取専用) に相当します。		
SERIALIZE <true false></true false>	ウィザードを連鎖化す るか(=TRUE)しない か(=FALSE)を指定し ます。ウィザードを連 鎖化する場合は、次の ウィザードの実行時に 前に入力された値が (iniファイルに)保存 されます。 NAMEプロパティ は、.iniのどのセクショ ンに値が格納されるか を決定します。 ブール型のプロパティ	SHALE=RE	デフォルトでは、この プロパティは 「FALSE」に設定され ます。
MODAL= <true false></true false>	ウィザードがモーダル か(=TRUE)否か (=FALSE)を定義しま す。		

252 トAssetCenter 4.2.0 - AssetCenterの高度な使い方

AssetCenter
表 14.3. ROOT ノードの物理プロパティ

プロパティ名=値	プロパティの説 明	例	コメント
IMAGE="<ビットマップファイル のパス>"IMAGE16="<ビットマッ プファイルのパス>"	ウィザードで表 示するピット マップタイプの グラフィック ファイル (.bmp)を定義 します。 文字列型のプロ パティ	IMAGE='Wizadomp'	値 場 示 定 、 和 フ ア イ ル の パ ス は 、 わ ア イ ル の パ ス に の 市 す の た の の た の で の た 、 れ ア イ ル の パ ス に の の た の で の た の で の た の で の た の で の た の で の た の で の た の で の た の で の た の で の た の で の た の で の た の で の た の で の た の で の た の で の た の た
WIDTH=<幅>	ウィザードの ウィンドウのデ フォルト幅 ( < 幅> )を定義し ます。値はtwip で指定します。 LONG ( 倍長整 数 ) 型のプロパ ティ	WIDTH=6000	

Acet	C
ASSAT	Lenter
7,0000	

プロパティ名=値	プロパティの説 明	例	コメント
HEIGHT=<高さ>	ウィザードの ウィンドウのデ フォルトの高さ (<高さ>)を定 義します。値は twipで指定しま す。 LONG(倍長整 数)型のプロパ	HEIGHT=5000	
MINWIDTH= <minwidth></minwidth>	ウィザードの ウィンドウの最 小幅を定義しま す。 値はtwipで指定 します。		
MINHEIGHT= <minheight></minheight>	ウィザードの ウィンドウの最 小の高さを定義 します。 値はtwipで指定 します。		
CTRLHEIGHT= <ctrlheight></ctrlheight>	高さが固定して いるコントロー ル(TEXTBOX コントロールな ど)の高さを定 義します。 間隔の値はtwip で指定します。		
LABELSPACING= <labelspacing></labelspacing>	コントロールの タイトルがコン トロールの上に ある場合、タイ トルと問の間隔 を定義します。 間隔の値はtwip で指定します。		

			AssetCenterの高度な使い方
プロパティ名=値	プロパティの説 明	例	コメント
CTRLSPACING= <ctrlspacing></ctrlspacing>	2つのコント ロール間の上下 の間隔を定義し ます。		
	間隔の値はtwip で指定します。		
IMGBORDER= <width></width>	ウィザードの画 像とコントロー ル間の左右の間 隔を定義しま す。		
	値はtwipで指定 します。		
NAVIGATION= <true false></true false>	ウィザードウィ ンドウ内の[次 へ]や[キャン セル]ボタンを 含むナビゲー ションバーを表 示するか (=TRUE)しな いかを (=FALSE)指 定します		
CONFIRMCANCEL= <true false></true false>	たりより。 キャンセルの確 認メッセージを 表示するか (=TRUE)しな いか (=FALSE)を 定義します		
DEFAULTONNEXT= <true false></true false>	<ul> <li>デフォルト         <ul> <li>デフォルト</li> <li>(=TRUE)で</li> <li>[次へ]ボタン</li> <li>を選択します。</li> </ul> </li> <li>DFALIONEXEFASE         <ul> <li>の場合、デフォ</li> <li>ルトで選択され</li> <li>るボタンは[終 了]です。</li> </ul> </li> </ul>		

## ROOTノードのサブノード

ROOTノードに定義できるサブノードのタイプを下の表に示します。各ノードタ イプはオブジェクトを表します。

#### 表 14.4. ROOT ノードのサブノード

ノードタイプ	説明
PAGE	ウィザードのページ名。
FINISH	ウィザードの最終ページからのトランジ
	ション名(終了)。このトランジションタ
	イプのノードには「FROM」および「TO」
	プロパティはありません。
START	例えば、ウィザードの起動時に(「DO」
	プロパティ)を使って実行するスクリプト
	およびウィザードの開始ページ(「TO」
	プロパティ)名が含まれます。
PARAMS	FINISHサブノードのプロパティCHAINが
	入力されていると、ウィザードのパラメー
	タを別のウィザードへ転送できます。
TIMER	ウィザードのページへタイマを関連付ける
	ことができます。

## PAGE ノードの定義

PAGEノードでは、ウィザードのページ名を定義します。このノードは、この ノードに適用できるプロパティブロックと、すべてのサブノード(ページで定 義されたオブジェクトを定義するサブノードのセット)で構成されます。

## PAGE ノードのシンタックス

PAGEノードのシンタックスは次の通りです。

' ページの宣言 { Page <ページ名> ' PAGE ノードのプロパティブロック IMAGE=... TITLE=...

```
└PAGEノードのサブノードの定義
{ TRANSITION
...
}
{ <コントロールのタイプ> <コントロール名>
...
}
....
```

# PAGE ノードのプロパティ

PAGEノードで定義できる全プロパティの一覧は、次の表の通りです。

### 表 14.5. PAGE ノードの論理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
TITLE="<ページの夕 イトル>"	ページのタイトルを 定義します。このタ イトルは、ページ上 部に太字で表示され ます。 文字列型のプロパ ティ	TITLE="移動"	このプロパティに値 を定義しない場合 は、ROOTノードの 「TITLE」プロパ ティの値が継承され ます。ラベルと異な り、この文字列は HTMLをサポートし ません。
ONENTER=<スクリ プト>	[次へ]または[前 へ]ボタンをクリッ クしてページにアク セスする際、実行さ れるBasicスクリプト を定義します。 文字列型のプロパ ティ	{ONENTER = AmMsgBox ("Hello") }	

#### 表 14.6. PAGE ノードの物理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
IMAGE="<ビット マップファイルのパ ス>" IMAGE16="<ビット マップファイルのパ ス>"	ウィザードのページ で表示するビット マップタイプのグラ フィックファイル (.bmp)を定義しま す。 文字列型のプロパ ティ	IMAGE =" C: /Images /Page1.bmp"	このプロパティに値 を定義しない場合 は、ROOTノードの IMAGEプロパティの 値が継承されます。 このプロパティに空 の値を定義すると、 画像は表示されません。 「IMAGE16」に値を 定義すると、画面の 色深度が16の場合、 「IMAGE」でなくこ のプロパティが使わ
			100 7.

## PAGE ノードのサブノード

PAGEノードには、2つのタイプのサブノードを定義できます。

#### 表 14.7. PAGE ノードのサブノード

ノードタイプ / オブジェクト	説明
<コントロール タイプ> <コントロール名>	現在のページに表示するコントロールを定
	義します。
	ウィザードの現在のページと別のページ間
	のトランジション名を定義します。
TIMER	ウィザードのページヘタイマを関連付ける
	ことができます。

## TRANSITION ノードの定義

TRANSITIONノードでは、ウィザード内の2つのページ間の移動を定義します。 このノードは、プロパティのブロックだけで構成されます。

### 💋 注意:

トランジションはPAGEノード内から定義(この場合、「FROM」プロパティは 必要ありません)またはROOTノード内から定義できます。ウィザードを閉じる 最終トランジションは、FINISHノード(ROOTノードの)で定義し、「FROM」 および「TO」プロパティは必要ありません。

# TRANSITION J – Foby System 7

TRANSITIONノードのシンタックスは次の通りです。

¦ トランジションの宣言 { TRANSITION0 <トランジション名> ' TRANSITION ノードのプロパティブロック FROM=... TO=... CONDITION=...

# TRANSITION ノードのプロパティ

次の表に、TRANSITIONノードで定義できる全プロパティの一覧を表示します。

#### 表 14.8. TRANSITION ノードの論理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
FROM="<トランジ ションのリンク元の ページの名前>"	トランジションのリ ンク元のページ名を 定義します。 文字列型のプロパ ティ	FROM="Page2"	このプロパティは、 トランジションが ROOTノード内で定 義されている場合は 必須、PAGEノー ド、FINISHノードま たはSTARTノードで 定義されている場合 は適用されません。

プロパティ名=値	プロパティの説明	例	コメント
TO="<ターゲット ページの名前>"	トランジションの ターゲットページ名 を定義します。 文字列型のプロパ ティ	TO="Page3"	このプロパティは、 ROOTノードまたは PAGEノード内で定 義されている場合は 必須、FINISHノード で定義されている場 合は適用されませ ん。
CONDITION=<スク リプト>	トランジションを起 動するのに必要な条 件を定義します。 ブール値を返すスク リプトタイプのプロ パティ	CONDITION= {Comment}="user"	このプロパティは、 STARTノードでは使 用できません。
DO=<スクリプト>	トランジション時に 実行されるスクリプ トを定義します。 文字列型のプロパ ティ	{DO= Filter=""}	

## TRANSITION ノードの特徴

TRANSITIONノードには、サブノードはありません。

### ROOTノードでトランジションを定義する理由

PAGEノードの外でトランジションを定義すると、すべてのウィザードで再使用 可能なページを作成できるため、スクリプトの作成を簡略化できます。

### FINISH ノードの定義

FINISHノードでは、ウィザードの最終ページにリンクする最終トランジション を定義します。これは、「FROM」および「TO」プロパティを持たない特別な タイプのTRANSITIONノードです。その他の点では、FINISHノードのシンタッ クスとプロパティは、TRANSITIONノードと同じです。

FINISHノードに固有のCHAINプロパティを使うと、他のウィザードの実行をト リガできます。

表 14.9. FINISH ノードの論理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
CHAIN=<実行する ウィザードのSQL名 >	現在のウィザードの 終了時に実行する ウィザードのSQL名 を定義します。	CHAIN="Move"	
	このプロパティが入 力されていないと ウィザードは実行さ れません。		
	文字列型のプロパ ティ		
CONDITION=<スク リプト>	[ <b>終了]</b> ボタンを有 効にする条件を定義 します。		
	文字列型のプロパ ティ		
D0=<スクリプト>	ウィザードの最後に 実行するスクリプト を定義します。		
	ブール値を返すスク リプトタイプのプロ パティ		

### 💋 注意:

PARAMSノードを使うと、次のウィザードにパラメータを渡すことができます。

### 表 14.10. FINISH ノードの物理プロパティ

プロパティ名=値	プロパティの説明	例
SUMMARY= <true false></true false>	ウィザード実行時に概要の	
	ページを表示するか	
	(=TRUE)しないか	
	(=FALSE)を定義します。	
	amLog関数とamProgress関数	
	によりこのページに入力で	
	きるようになります。	

#### AssetCenter

S-OMPROCRESSBAR-TRUFFALSE>	概要ページに進行状況バー を表示するか(=TRUE)し ないか(=FALSE)を定義し ます。
SHOWLOGLIST= <true false></true false>	概要ページに進行状況ログ を表示するか(=TRUE)し ないか(=FALSE)を定義し ます。
LABEL="タイトル"	概要ページにタイトルを表 示します。
ISHTML= <true false></true false>	ラベルのテキストの属性を 定義します。 ブール型のプロパティ
TITLE="タイトル"	概要ページのタイトルを定 義します。 概要ページのデフォルトの
	低安ベージのアンオルドの タイトルは、ROOTノード のタイトルです。

# START ノードの定義

STARTノードでは、ウィザードの開始方法を定義します。これは、「FROM」 または「CONDITION」プロパティを持たない特殊なタイプのTRANSITIONノー ドです。その他の点では、STARTノードのシンタックスとプロパティは、 TRANSITIONノードと同じです。

### 表 14.11. START ノードの論理プロパティ

プロパティ名=値	プロパティの説明	例
DO=<スクリプト>	起動時に実行されるスクリ プトを定義します。	
	ブール値を返すスクリプト タイプのプロパティ	
TO="<開始ページ名>"	表示する最初のページの名 前を定義します。 文字列型のプロパティ	

💋 注意:

このノードが存在しない場合、ウィザードは最初のページから起動します。

# TIMER ノードの定義

TIMERノードは、定期的にタスクを実行できるようにします。

#### 表 14.12. TIMER ノードの論理プロパティ

プロパティ名=値	プロパティの説明	例
AUTO= <true false></true false>	ページの表示時にタイマが 自動的に起動されるかどう かを指定します。	
	このプロパティは、タイマ を停止または再起動するた めにも使用されます。	
	ブール型のプロパティ	
ENABLED= <true false></true false>	タイマをアクティブにする か(=TRUE)しないか (=FALSE)を定義します。	
	ブール型のプロパティ	
INTERVAL=間隔	タイマを実行する時間の間 隔を定義します。	
	間隔はミリ秒単位で指定し ます。	
TIMER="スクリプト"	タイマの間隔 ("INTERVAL")ごとに実 行される内容を定義しま す。	
	スクリプトタイプのプロパ ティ	
VALUE=tickcount	時間の間隔が経過した回 数。間隔に依存するプロパ ティは全て、定期的な間隔 で再評価されます。	
	tickcountの値は自動的に増 加します。	

## 倍長整数および文字列ノードの定義

LONG(倍長整数)およびSTRING(文字列)ノードでは、変数を定義します。 変数はウィザードのすべてのノードで参照できます。ノード名によって、変数 名が決まります。

これらのノードには1つのプロパティしかなく、そのデータ型はノードに依存します。データ型は倍長整数ノードの場合は、倍長整数、文字列ノードの場合は、 文字列です。この「VALUE」プロパティを使って、変数の値を定義できます。

#### 表 14.13. 倍長整数および文字列ノードの論理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
VALUE=<値>	<ul> <li>ノード名を名前に持</li> <li>つ変数の値を定義します。</li> <li>倍長整数の場合は倍</li> <li>長整数型、文字列</li> <li>ノードの場合は文字</li> <li>列型</li> </ul>	VALUE=12	

💋 注意:

倍長整数および文字列ノードは、ウィザードのどのノードでも定義できます。 サブノードはありません。

## コントロールノードの定義

ページのコントロールを使ってユーザは情報を指定できます。1ページ内に好き な数だけコントロールを定義できます。ページ内でのコントロールの編成は、 AssetCenterが自動的に管理します。定義した各コントロールの位置を指定する 必要はありません。

コントロールタイプのノードは、定義したコントロールに適用可能なプロパティ ブロックだけで構成されます

# コントロールノードの一般的なシンタックス

コントロールタイプのノードの一般的なシンタックスは次の通りです。

```
' コントロールの宣言
{ <コントロールのタイプ> <コントロール名>
' コントロールのプロパティ
...
```

# コントロールのタイプと関連プロパティ

すべてのコントロールには、共通のプロパティがあります。一方、特定のコン トロールに固有のプロパティもあります。

## 共通のプロパティ

すべてのコントロールに適用できるオプションのプロパティは、次の表の通り です。

### 表 14.14. すべてのコントロールに共通の論理プロパティ

プロパティの説明	例	コメント
トランジションを確 認するために、ユー ザにコントロールへ の入力を強制しま す。	MANDATORY=TRUE	このプロパティは CHECKBOXおよび LABELコントロール には使用できませ ん。
以下の場合このプロ パティは無視されま す。		
<ul> <li>コントロールが 表示されていな い。</li> </ul>		
<ul> <li>コントロールは 読取専用であ る。</li> </ul>		
<ul> <li>コントロールが 無効になってい る。</li> </ul>		
作成時にコントロー ルのデフォルト値を 定義します。<値>は そのコントロールに よって決まります。 このプロパティのタ イプはコントロール タイプによって変化 します(ブール、テ	例えば、 CHECKBOXコント ロールの場合、<値> はTRUEまたは FALSEになります	
	<ul> <li>ブロパティの説明</li> <li>トランジョンを確認したい</li> <li>認知したい</li> <li>ジションを確認したい</li> <li>ジションを確認したい</li> <li>ジントロールへの</li> <li>の人力を強いたい</li> <li>の場合このプロパティは</li> <li>オールが</li> <li>れたい</li> <li>コントロールが</li> <li>れたい</li> <li>コントロールは</li> <li>読取</li> <li>コントロールは</li> <li>ある。</li> <li>コントロールは</li> <li>ある。</li> <li>コントロールが</li> <li>ない</li> <li>ロールが</li> <li>シントロールが</li> <li>ない</li> <li>マントロールを</li> <li>たのすい</li> <li>マントロールを</li> <li>たのすい</li> <li>マントロールを</li> <li>たのすい</li> <li>マントロール</li> <li>マントローム</li> <li>マントロ</li></ul>	プロパティの説明       例         トランジションを確       MANDATORY=TRUE         認するために、ユー       ザにコントロールへ         ヴ、ロントロールへ       の入力を強制しま         す。       リ下の場合このプロ         パティは無視されま       す。         ・コントロールが       表示されていない。         ・コントロールは       読取専用である。         ・コントロールが       概効になっている。         作成時にコントロールが       例えば、         ルのデフォルト値を       CHECKBOXコント         定義します。<値>は       ロールの場合、         そのコントロールに       はTRUEまたは         よって決まります。       FALSEになります         このプロパティのタ       イプはコントロール         タイプによって変化       します(ブール、テ         キスト、など)       ー

プロパティ名=値	プロパティの説明	例	コメント
TERMANENT=TRUTALS>	ウィザードのある ページから別のペー ジへ移る時、コント ロールは削除されま す。 ウィザードのある ページから別のペー ジへ移る時に、コン トロールを保存し隠 すか(=TRUE)否か (=FALSE)を指定 します。		
	ブール型のプロパ ティ		
SERIALIZE-TRUEFALSE>	このプロパティで は、コントロールに 関してウィザードを シリアル化するかし ないかを指定できま す。ROOTノードの シリアル化が有効に なっている場合 (=TRUE)、コント ロールについてはシ リアル化を無効にす ることができます。		このプロパティは、 デフォルトでROOT ノードのSERIALIZE プロパティの値を含 みます。
HELP="ヘルプ"	このプロパティによ り、HTMLフォー マットのヘルプ用テ キストをウィザード のコントロール内に 含むことができま す。[Shifit+F1] キーを押してヘルプ を表示します。		

### 表 14.15. すべてのコントロールに共通の物理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
VISIBLE= <truefalse></truefalse>	コントロールを表示 するか ( =TRUE ) 、 しないか ( =FALSE ) を定義 します。 ブール型のプロパ ティ	Label1.Visible=TRUE	
ENABLED = <true false></true false>	コントロールをアク ティブにするか (=TRUE)、しない か(=FALSE)を定 義します。 ブール型のプロパ ティ	Choice1.Enabled=FALSE	
ENABLED = <true false></true false>	コントロールの値を 読み取り専用にして (=TRUE)、ユーザ が変更できないよう にするか、変更でき るようにするか (=FALSE)を定義 します。	READONLY=TRUE	
LABEL="<ラベルの テキスト>"	コントロールの上ま たは左に表示される 任意のテキストを定 義します。 文字列型のプロパ ティ	Choice1.Label="ユー ザの選択"	このラベルはHTML をサポートします。

プロパティ名=値	プロパティの説明	例	コメント
LABELLEFT	このプロパティは、 コントロールのラベ ルをコントロールの 左側に置きます。 このプロパティを使 用する場合は、 XOFFSETプロパティ に入力する必要があ		
	りょぅ。 ブール型のプロパ ティ		
XOFFSET	LABELLEFTプロパ ティを使用して、コ ントロールのラベル をコントロール左側 に置く場合、ラベル 用に確保される場所 を定義します。		
ISHTML	twip型のプロパティ ラベルのテキストの 属性を定義します。 デフォルトのテキス トの属性はHTMLで す。 ブール型のプロパ		このラベルはHTML をサポートします。
INDENT	ティ このプロパティは、 コントロールとラベ ルを右側に移動させ ます。		
YOFFSET	twip型のプロパティ コントロールとラベ ルの前の「オフセッ ト」を定義します。		
YOFFSET2	wipygのフロバディ コントロールとラベ ルの後の「オフセッ ト」を定義します。 twip型のプロパティ		

### CheckBoxコントロール

CHECKBOXコントロールでは、チェックボックスを定義します。

## プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、CHECKBOXコ ントロールでは、次のプロパティが認識されます。

#### 表 14.16. CHECKBOX コントロールのプロパティ

プロパティ名=値	プロパティの説明	例
CAPTION="<テキスト>"	チェックボックスのテキス トを定義します	TEXT="名前で識別"
	このテキストはHTMLでは ありません。1行のみを含ま なければなりません。	
	文字列型のプロパティ	

### ComboBoxコントロール

COMBOBOXコントロールでは、定義済みの値リスト(リストデータ)の1つの 選択肢を定義します。

## プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、COMBOBOXコ ントロールでは、次のプロパティが認識されます。

#### 表 14.17. COMBOBOXコントロールの物理プロパティ

プロパティ名=値	プロパティの説 明	例	コメント
VALUES="<名前=値, 名前=値, 名 前=値,>"	Comboコのして、 ローセージンでのした。 「し」では、 し」では、 し」では、 し」では、 し」で、 し」で、 し」で、 し」で、 し」で、 し」で、 し」で、 し」で	VALUES="Table of assets=asset, User=user"	値を省略する と、値が自動的 に割り当てられ ます。 <b>例</b> VALUES=A,B,C と定義すると、 VALUES=A=1B=2C=3 と値が割り当て られます。

## OptionButtonsコントロール

OPTIONBUTTONSコントロールでは、オプションボタン(ラジオボタン)のグ ループを定義します。

## プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、 OPTIONBUTTONSコントロールでは、次のプロパティが認識されます。

プロパティ名=値	プロパティの説明	例
VALUES="<タイトル=値, タ イトル=値, タイトル= 値,>"	CHOICEコントロールの値 の組み合わせ(「タイト ル」=「値」)を定義しま す。「タイトル」にはオプ ションボタンのテキストを 定義し、「値」には、ユー ザがこのオプションボタン を選択した場合にコント ロールに割り当てられる値 を定義します。 文字列型のプロパティ	VALUES="Table of assets=asset, User=user"
BORDER= <true false></true false>	オプションボタンのグルー プに境界を引くか (=TRUE)、引かないか (=FALSE)を指定します。 ボタンのグループが囲まれ ていると、テキストが枠の 端に統合されます。このテ キストはHTMLおよび多言 語であってはなりません。 ブール型のプロパティ	BORDER= TRUE

#### 表 14.18. OPTIONBUTTONSコントロールの物理プロパティ

## ListBoxコントロール

LISTBOXコントロールでは、選択可能なオブジェクトのリストを定義します。 LISTBOXコントロールは、複数の列にすることができます。

# プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、LISTBOXコン トロールでは、次のプロパティが認識されます。

### 表 14.19. LISTBOXコントロールの物理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
LISTHEIGHT = < パーセント>	LISTBOXコントロー ルのサイズを、ウィ ザード全体に使われ ている他のLISTBOX コントロールを基準 して定義します。 LONG(倍長整数) 型のプロパティ	LISTHEIGHT=50	値がそれぞれ「10」 と「20」の2つの LISTBOXコントロー ルがある場合、2番 目のコントロールの 高さは最初のコント ロールの2倍になり ます。
MULTISEL = <true false></true false>	コントロールで複数 選択をサポートする (=TRUE)かしない (=FALSE)かを指 定します。 ブール型のプロパ ティ	MULTISEL=TRUE	
COLTITLE = "<列 列  列>"	リスト内の列のタイ トルを定義します。 「列」には列のテキ ストを定義します。 文字列型のプロパ ティ	COLTITLE = "姓 名 前"	
COLWIDTH = "<幅  幅 幅>"	コントロール全体の サイズに比例した列 のサイズを定義しま す。 文字列型のプロパ ティ	COLWIDTH = "50 50"	

プロパティ名=値	プロパティの説明	例	コメント
VALUES = "<テキス ト テキスト =値, テ キスト テキスト = 値,>"	LISTBOXコントロー ルの値の組み合わせ (テキスト テキスト  =値)を定義しま す。「テキスト テキ スト 」には、 LISTBOXコントロー ルの行の各列に表示 するテキストを定義 します。「値」に は、ユーザがこの行 を選択した場合にこ のコントロールに割 り当てられる値を定 義します。 文字列型のプロパ ティ	VALUES="Table of assets=asset, , User=user,"	値を省略すると、値 が自動的に割り当て られます。 VALUES=A,B,Cと定 義すると、 VALUES=A=1,B=2,C=3 と値が割り当てられ ます。 <b>例</b> このプロパティに は、直接入力した り、次のように AmdbGetList関数を 使って入力すること もできます。 VALUES = AmDbGetList("SELECT Name, FirstName FROM amEmpIDept WHERE Name Like 'A%'", " ", ",", "=") 「VALUES」プロパ ティを「VALUE」 プロパティと混同し ないでください。
EDITABLE="<0 1>"	列のテキストが編集 可能か(=TRUE)、 不可能か (=FALSE)を定義 します。 文字列型のプロパ ティ	EDITABLE="0 1"	
TABLE="<テーブル 名>"	列のタイトルが COLNAMEプロパ ティで定義されてい る場合、列のタイト ルの適用コンテキス トを定義します。	TABLE="amEmplDept"	

プロパティ名=値	プロパティの説明	例	コメント
COLNAME="<名前  名前 名前>"	リストの列のプロパ ティとタイトルを、 フィールドのSQL名 と共に定義します。 TABLEプロパティに は値を入力する必要 があります。 列のタイトル用に 「名前」をSQLで置 き換えます。	CCLNAVVE="NameFitNametHe"	
	列のタイトルに COLTITLEプロパ ティで値が入力され ている場合、このプ ロパティは COLNAMEプロパ ティよりも優先され ますが、SQLフィー ルドタイプ(テキス ト、日付など)を保 持します。		
MULTISEL = <true false></true false>	リスト用に複数選択 の使用を定義しま す。 ブール型のプロパ ティ	MULTISEL=1	

### 表 14.20. LISTBOX コントロールのメソッド

プロパティ名=値	プロパティの説明	例	コメント
VALUES (i)	(i) 行目の内容を返	a b c	
	します。		
VALUES.COUNT()	VALUESプロパティ	retval =	
	内の行数を計算しま	{listbox1.values.count()}	
	す。		
VALUES.CELL(h,v)	座標 ( 水平座標,垂直 座標 ) で指定された セルの内容を返しま す。	VALUES.CELL(2,4)	

プロパティ名=値	プロパティの説明	例	コメント
VALUES.COLUMN(i)	列(i)の内容(値) を返します。	VALUES.COLUMN(1)	
	i=0の場合または空の 場合、この命令は列 の識別子(ID)を返 します。		
VALUES.SORT(iCol,	列 ( i ) を昇順または	{ LISTBOX lb	
bAsc)	降順 ( bAsc=1または	VALUES = "第1,第2,	
	bAsc=0)で並べ替え	第3" }	
	ます。	{	
		COMMANDBUTTON	
		btn { CLICK = RetVal	
		= {Ib.Values.Sort(1)}	
		} }	
VALUES (i,0)	行 (i) のID値を返し		
ID(i)	ます。		

### 表 14.21. LISTBOX コントロールの必須論理プロパティ

TABLE=<テーブル名> 列のタイトルの抽出に使う TABLE=amAsset テープル名。 文字列型のプロパティ	
ンジョン シンクロック シンクロック シンクロック シンクロック シンクロック シンクロ シンクロ シンク	
COLNAME=<タイトル タイ	ame"

## Labelコントロール

LABELコントロールでは、ラベルを定義します。このコントロールには次のプ ロパティがあります。

表 14.22. LABELコントロールの物理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
CAPTION=<テキス ト>	ラベル内に表示され るテキストが含まれ ます。	CAPTION="場所の選 択"	

## ProgressBarコントロール

PROGRESSBARコントロールでは、進行状況バーを定義します。

## プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、PROGRESSBAR コントロールでは、次のプロパティが認識されます。

#### 表 14.23. PROGRESSBAR コントロールの物理プロパティ

プロパティ名=値	プロパティの説明	例
MAXVALUE=<最大值>	進行状況バーの100%に対応 する最大値を定義します。	MAXVALUE=200
	「VALUE」プロパティは、 コントロールの現在の値を 示します。	
	LONG(倍長整数)型のプ ロパティ	

## CommandButton コントロール

COMMANDBUTTONコントロールでは、アクションボタンを定義します。

# プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、 COMMANDBUTTONコントロールでは、次のプロパティが認識されます。

表 14.24. COMMANDBUTTONコントロールの物理プロパティ

プロパティ名=値	プロパティの説明	例
WIDTH=<幅>	ボタンの幅をtwipで定義し ます。	WIDTH=250
	LONG(倍長整数)型のプ ロパティ	
HEIGHT=<高さ>	ボタンの高さをtwipで定義 します。	HEIGHT=125
	LONG(倍長整数)型のプ ロパティ	
CAPTION=<テキスト>	ボタン内に表示するテキス ト(非HTML)を定義しま す。	CAPTION="開始"
	文字列型のプロパティ	
CLICK= <basicスクリフト></basicスクリフト>	ユーザがこのボタンをク リックしたときに実行され るBasicスクリプトを定義し ます。	

## DBListBoxコントロール

DBLISTBOXでは、データベースから選択できるレコードのリストを定義しま す。このコントロールは、複数の列にすることができます。コントロールに表 示されるリストは、AssetCenterデータベースで部分AQLクエリ(WHERE句のみ 使用)を実行した結果です。

### 💋 注意:

「VALUE」プロパティは、選択した行の識別子(Id)のリストを返します。リ ストのセルの値にアクセスすることはできません。セル値にアクセスするには、 別のクエリを実行するか、LISTBOXタイプのコントロールを使う必要がありま す。

# プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、DBLISTBOXコ ントロールでは、次のプロパティが認識されます。

表 14.25. DBLISTBOXコントロールの物理プロパティ

プロパティ名=値	プロパティの説明	例	コメント
TABLE="<テーブル のSQL名>"	クエリを行うテーブ ルを定義します。 文字列型のプロパ ティ	TABLE=amAsset	必須プロパティ
COLNAME="< フィールドまたはリ ンクのSQL名 フィー ルドまたはリンクの SQL名 >"	データベースから抽 出するデータ項目を 定義します(SQL名 で識別)。 文字列型のプロパ ティ	COLNAME = "Name FirstName"	
COLWIDTH = "<幅  幅 幅 >"	データベースの列の 幅を、DBLISTBOX コントロール全体に 占めるパーセントで 定義します。 文字列型のプロパ ティ	COLWIDTH="40 60"	
LISTHEIGHT = < パーセント>	DBLISTBOXコント ロールのサイズを、 ウィザード全体の他 のDBLISTBOXコン トロールを基準にし て定義します。 LONG(倍長整数) 型のプロパティ	LISTHEIGHT=50	このプロパティの値 がそれぞれ「10」と 「20」の2つの 「DBLISTBOX」コ ントロールがある場 合、2番目のコント ロールの高さは最初 のコントロールの2 倍になります。
TREE= <true false></true false>	データをツリーで表 示 ( =TRUE ) する か、表示しない ( =FALSE ) かを指 定します。 ブール型のプロパ ティ	TREE=TRUE	このプロパティはデ フォルトでは 「FALSE」に設定さ れます。

プロパティ名=値	プロパティの説明	例	コメント
MULTISEL = <true false></true false>	コントロールで複数 選択をサポートする (=TRUE)かしない (=FALSE)かを指 定します。 ブール型のプロパ ティ	MULTISEL=TRUE	
MULTISEL = <true false></true false>	このプロパティを TRUEに設定する場 合、AssetCenter行上 をダブルクリックす ると、現在のページ の「次へ」ボタンの クリックがシミュ レートされます。	DBLCLICK=FALSE	
FILTER = " <condition>"</condition>	クエリで処理するレ コードをフィルタで 検索するAQLの 「WHERE」条件を 定義します。 文字列型のプロパ ティ	FILTER = "User.IEmpIDeptId = 'Colombo, Gerard' "	
MAXSEL = <true false></true false>	99よりも多くの選択 できるか(=TRUE) できないか (=FALSE)を定義 します。 デフォルトで (=TRUE)選択は99 要素に制限されてい ます。		
VALUES.SORT(iCol, bAsc)	列(iCol)を昇順ま たは降順(bAsc=1ま たはbAsc=0)で並べ 替えます。	VALUES.SORT(2, 0)	

280 > AssetCenter 4.2.0 - AssetCenterの高度な使い方

AssetCenter

<mark>プロパティ名=値</mark> [Value.]ISSELECTION()	<b>プロバティの説明</b> 選択が0以外の値を 含む場合、つまり ユーザがリストから 何も選択しない場 合、このメソッドは 非NULL値を返しま す。	例 MANDATORY = not {dblistbox1.lsSelection()} (ユーザがリストから何も選択しない と、トランジション は不可能になります。)	<u>コメント</u>
TABLELABEL([NameType])	このメソッドはコン トロールが定義され るテーブルのタイト ルを返します。 タイトルタイプ (iNameType)は以 下の通りです。 • 1 - システム名 • 2 - SQL名 • 3 - デフォルトの タイトル • 4 - 説明(ヘルプ 用テキスト) TABLEプロパティに は値が入力されなけ ればなりません。	RetVal = {dblistbox1.table.label(2)}	

## DBQueryBoxコントロール

DBQUERYBOXコントロールでは、選択可能なレコードのリストを定義します。 このコントロールは、複数の列にすることができます。コントロールに表示さ れるリストは、AssetCenterデータベースの完全AQLクエリ(WHERE句のみ使 用)を実行した結果です。

# プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、DBQUERYBOX コントロールでは、次のプロパティが認識されます。

プロパティ名=値	プロパティの説 明	例	コメント
QUERY="<完全AQLクエリ>"	DBLISTコント ロールに表示す る情報を返す AQLクエリを定 義します。 文字列型のプロ パティ	QUERY="SELECT Name, FirstName FROM amEmpIDept WHERE Location='大阪ビ ル'"	
COLTITLE="<列 列 >"	リストの列のタ イトルを定義し ます。 文字列型のプロ パティ	COLTITLE="姓  名前"	
COLWIDTH="<幅 幅 >"	リストの列のサ イズを、コント ロール全体のサ イズにとかる パーセントで定 義します。 文字可のプロ パティ	COLWIDTH = "50 50"	
LISTHEIGHT=<パーセント>	DBQUERYBOX コントロールの サイズを、ウィ ザード全体の他 の DBQUERYBOX コントロールを 基準にして定義 します。 LONG(倍長整 数)型のプロパ ティ	LISTHEIGHT=50	このプロパティ の値がそれぞれ 「10」と「20」 の2つの DBQUERYBOX コントロールが ある場合、2番 目のコントロー ルの高さは最初 のコントロール の2倍になりま す。

### 表 14.26. DBQUERYBOXコントロールの物理プロパティ

		Asse	etCenterの高度な使い方
プロパティ名=値	プロパティの説 明	例	コメント
TREE= <true false></true false>	データをツリー で表示 (=TRUE)する か、表示しない (=FALSE)か を指定します。 ブール型のプロ パティ	TREE=TRUE	このプロパティ はデフォルトで は「FALSE」に 設定されます。
MAXSEL = <true false></true false>	99よりも多くの 選択できるか (=TRUE)でき ないか (=FALSE)を 定義します。 デフォルトで (=TRUE)選択 は99要素に制限 されています。		
MULTISEL= <true false></true false>	コントロールで 複数選択をサ ポートする (=TRUE)かし ない (=FALSE)か を指定します。 ブール型のプロ パティ	MULTISEL=TRUE	
DBLCLICK= <true false></true false>	このプロパティ をTRUEに設定 する場合、 AssetCenter行上 をダブルクリッ クすると、現在 のページの「次 ヘ」ボタンのク リックがシミュ レートされま す。 プール型のプロ パティ	DBLCLICK=FALSE	

プロパティ名=値	プロパティの説 明	例	コメント
[Value.]ISSELECTION()	選択が0以外の 値を含む場合、 つまりユーザが リストから何も 選んしない場 くは非NULL値 を返します。		

### DBEdit コントロール

DBEDITコントロールでは、AssetCenterデータベースのフィールドに入力するの に使ったのと同じコントロールが作成されます。コントロールは、各フィール ドのデータ型(日付、金額など)によって異なります。

### 💋 注意:

このコントロールの拡大鏡ボタン Na を使って、データベースから有効な値を選 択できますが、別の値を入力することもできます。 このコントロールでは、「VALUE」プロパティは「可変」です(コントロール によって異なります)。

### プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、DBEDITコント ロールでは、次のプロパティが認識されます。

#### 表 14.27. 標準モードのDBEDITコントロールの物理プロパティ

プロパティ名=値	プロパティの説明	例
TABLE="<テーブルのSQL名	開始テーブルのSQL名	TABLE=amAsset
>"	文字列型のプロパティ	
FIELD="<フィールドのSQL	コントロールに使用される	FIELD="seAcquMethod"
名>"	フィールドのAQL名。	
	文字列型のプロパティ	

プロパティ名=値	プロパティの説明	例
TABLE.LABEL([iNameType])	このメソッドはテーブルの タイトルを返します。	本章の「DBQUERYBOX <b>コ</b> ントロールの物理プロパ
	タイトルタイプ	ティ」の節を参照してくだ
	(iNameType)は以下の通	さい。
	りです。	
	<ul> <li>1-システム名</li> <li>0.00 名</li> </ul>	
	<ul> <li>2-SQL名</li> <li>2 デフォリトのタイト</li> </ul>	
	• 3-ナノオルトのタイト ル	
	• 4-説明(ヘルプ用テキ	
	スト)	
	TABLEブロバティとNAME	
	れなければなりません。	
FIELD.LABEL([iNameType])	このメソッドはフィールド	
	のタイトルを返します。	
	タイトルタイプ	
	(iNameType)は以トの通 りです。	
	<ul> <li>1-システム名</li> </ul>	
	• 2 - SQL名	
	• 3-デフォルトのタイト	
	<ul> <li>4-説明(ヘルブ用テキ スト)</li> </ul>	
	TABLEプロパティには値が	
	入力されなければなりませ ,	
	N.	

## DBTableコントロール

DBTABLEコントロールは、AssetCenterデータベースのテーブルに入力するためのコントロールを作成します。

# プロパティ

このコントロールに固有のプロパティはありません。

### DBPath コントロール

DBPATHは、AssetCenterデータベースのパスを入力するためのコントロールを 作成します。

## プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、DBPATHコン トロールでは、次のプロパティが認識されます。

#### 表 14.28. DBPATHコントロールの必須論理プロパティ

プロパティ名=値	プロパティの説明	例
TABLE="<テーブルのSQL名	パスを選択する際の選択元	TABLE=amAsset
>"	のテーブル名。	
	文字列型のプロパティ	

### LinkEditコントロール

LINKEDITコントロールは、AssetCenterデータベースにリンクを入力するための コントロールを作成します。

### プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、LINKEDITコン トロールでは、次のプロパティが認識されます。

#### 表 14.29. LINKEDITコントロールの論理プロパティ

プロパティ名=値	プロパティの説明	例 / コメント
TABLE=<テーブルのSQL名	リンクを選択するテーブル	TABLE=amAsset
>"	名。	
	文字列型のプロパティ	
FILTER= <aqlクエリの< td=""><td>AQLフィルタを定義しま</td><td>このプロパティはオプショ</td></aqlクエリの<>	AQLフィルタを定義しま	このプロパティはオプショ
WHERE句>	す。	ンです。
	文字列型のプロパティ	

プロパティ名=値	プロパティの説明	例/コメント
LINK=<リンクのSQL名>"	TABLEプロパティ内で定義 されるテーブルリンクの SQL名。	LINK="POrLIne"
	オプションプロパティ	
ZOOM= <true false></true false>	拡大鏡ツールを表示するか (=TRUE)しないか (=FALSE)を定義します。	
	ウィザードがモーダルでな い(ROOTノードでプロパ ティMODAL=FALSE)場合 のみこのプロパティが適用	
SRCCHOICE= <true false></true false>	<ul> <li></li></ul>	
	ウィザードがモーダルでな	
	い ( ROOT ノードでプロパ	
	ティMODAL=FALSE)場合	
	のみこのプロパティが適用 さわます	
TABLE LABEL ([iNameType])	<u>このメソッドは リンクの</u>	本音の「DBOLIERYBOXコ
	ソーステーブルのタイトル	ントロールの物理プロパ
	を返します。	ティ」の節を参照してくだ
	タイトルタイプ	さい。
	(iNameType)は以下の通	
	りです。	
	<ul> <li>1-システム名</li> </ul>	
	• 2-SQL名	
	<ul> <li>3-デフォルトのタイト ル</li> </ul>	
	<ul> <li>4 - 説明(ヘルプ用テキ スト)</li> </ul>	
	TABLEプロパティには値が 入力されなければなりませ ,	
	<i>n</i> o.	

AssetCenter

プロパティ名=値	プロパティの説明	例 / コメント
LINK.LABEL([iNameType])	このメソッドはリンクのタ	
	1トルを返します。	
	タイトルタイプ	
	(iNameType)は以下の通	
	りです。	
	<ul> <li>1-システム名</li> </ul>	
	• 2 - SQL名	
	• 3-デフォルトのタイト	
	ル	
	• 4-説明(ヘルプ用テキ	
	スト)	
	TABLEプロパティとLINKプ	
	ロパティには値が入力され	
	ていなければなりません。	

## TextBoxコントロール

TEXTBOXコントロールは、テキストを入力するためのコントロールを作成します。

## プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、TEXTBOXコン トロールでは、次のプロパティが認識されます。

#### 表 14.30. TEXTBOXコントロールの物理プロパティ

プロパティ名=値	プロパティの説明	例
MULTILINE=<数値>	TEXTBOXコントロールが1 行の場合、このプロパティ は「0」に、複数行の場合 は、コントロールの表示高 さのパーセント値になりま す。	MULTILINE=50
PASSWORD= <true false></true false>	タイプされたテキストをマ スクするか(=TRUE)しな いか(=FALSE)を定義しま す。	
# CHARTコントロール

CHARTコントロールではグラフを表示できます。グラフを複数の系列で構成す ることも可能です。

# プロパティ

すべてのコントロールに共通のオプションのプロパティに加え、CHARTコント ロールでは、次のプロパティが認識されます。

#### 表 14.31. CHARTコントロールの論理プロパティ

プロパティ名= 値	プロパティの説明	例
SERIES	縦線(パイプ文字)で区切られた 系列のリストを定義します。	SERIES="購入価格 販売価格"
	このリストは空であってはなりま せん。このリストなしではグラフ	
	は表示されません。	
VALUES	グラフの系列の数値を定義しま す。	VALUES="1 2,1 4"
	2次元の図表	
FORMAT	データ型を定義します。	
	<ul> <li>倍長整数(long)</li> </ul>	
	• 倍精度浮動小数点型(double)	
	• 数值(number)	
	<ul> <li>パーセント (percent)</li> </ul>	
SERIE	クリックした系列の番号	
	CHARTプロパティは相互作用モー	
	ド ( =TRUE ) でなければなりませ	
	ん。	
INDEX	クリックした列の番号	
	CHARTプロパティは相互作用モー	
	ド ( =TRUE ) でなければなりませ	
	ん。	
CLICK	図表上でクリックする際にこのプ	
	ロパティのスクリプトを呼び出し	
	ます。	

#### 表 14.32. CHARTコントロールの物理プロパティ

プロパティ名= 値	プロパティの説明	例
MODE	図表のタイプを定義します。	
	• MODE=0:縦の棒グラフ	
	• MODE=1:横の棒グラフ	
	• MODE=3:円グラフ	
LABELS	列のタイトルを定義します。	1月 2月
3D	グラフを3Dにするか(=TRUE)	
	しないか(=FALSE)を定義しま す。	
COLORS	各系列用の色のリストを定義しま	255 16777215 16711680
	す。リストの要素は縦線(パイプ	赤、白、青色を表示します。
	文字)で区切られます。	
	RGB値は十進法です。	
INTERACTIVE	グラフが相互作用するか	
	(=TRUE)しないか	
	(=FALSE)、つまりグラフがマ	
	ワスのホインタの変化に応じて起	
	_ <u></u>	
POPUP	小ツノアツノメニューを衣示9る か(_TPUE)したいか	
	ハ (=IROE) ひないか (=FAISE)を定義します	
BACKGROUND	グラフに背景を表示するか	
Briefferterte	(=TRUE)しないか(=FALSE)	
	を定義します。	
BACKIMAGE	グラフの背景になる画像のパスを	
	定義します。	
	背景を表示するには、	
	BACKIMAGEプロパティが有効	
	(=TRUE)でなければなりませ	
	<i>h</i> 。	
STACKED	積み上げ棒グラフにするか	
	(=TRUE)しないか(=FALSE)	
CHARTHEIGHT	CHARIコントロールの相対的な	
	サイスを、ワイサート全体の他の	
	コノドロールを基件にして正我します	
CAPTION	 タイトルを表示します	

<sup>290</sup> トAssetCenter 4.2.0 - AssetCenterの高度な使い方

AssetCenterの高度な使い方

プロパティ名= 値	プロパティの説明	例
ELEVATION	3D円グラフの仰角(傾き)度を定 義します。	
ROTATION	円グラフの回転角度を定義しま す。 値は「度」単位で指定します。	
DISPLAYLABELS	列のタイトル(LABELS)を表示 するか(=TRUE)しないか (=FALSE)を定義します。	
DISPLAYSLEGEND	系列の凡例を表示するか (=TRUE)しないか(=FALSE) を定義します。	

# FILEEDIT**コントロール**

このコントロールはダイアログボックスを表示し、ファイルやフォルダの保存 または読み込みを可能にします。

表 14.33. FILEEDITLコントロールのプロパティ

プロパティ名=値	プロパティの説明	例
OPENMODE	ダイアログボックスのタイプを定義しま す。	
	• OPENMODE=1:ファイルを開きます。	
	• OPENMODE=2:ファイルを保存しま す。	
	• OPENMODE=4:フォルダを開きます。	
	<ul> <li>OPENMODE=8:フォルダを保存しま</li> </ul>	
	ッ。 ダイアログボックフロに列送されるファイ	(* tyt) * tyt (* con) * con
FILTERS	リーンシングにの手を行っているファイルの表示基準を定義します。	
DEFEXT	デフォルトのファイルの拡張子を定義しま す。	(*.scn) *.scn

# TICKEDITコントロール

このコントロールではスケジューラを挿入できます。

#### 表 14.34. TICKEDITコントロールのプロパティ

プロパティ名=値	プロパティの説明	例
VALUE	ユーザがスケジューラ内で定義したパラ	
	メータを、文字列の形式で表示します。	
LISTHEIGHT	TICKEDITコントロールのサイズを、ウィ	
	ザード全体の他のコントロールを基準にし	
	て定義します。	
	LONG(倍長整数)型のプロパティ	

## CALENDAR コントロール

このコントロールではカレンダを挿入できます。

## TIMESPANEDIT**コントロール**

このコントロールでは、期間の入力ゾーンを挿入できます。

### NUMBOXコントロール

このコントロールでは数値タイプのコントロールを挿入できます。

#### 表 14.35. NUMBOXコントロールのプロパティ

プロパティ名=値	プロパティの説明	例
MINVALUE	数の最小値を定義します。デフォルト値は 無限値です。	
MAXVALUE	数の最大値を定義します。デフォルト値は 無限値です。	
FORMAT	数のフォーマットを定義します。 ・ オペレーティングシステムのコントロー ルパネル内で定義したフォーマットの LONG ・ RAWLONG ・ オペレーティングシステムのコントロー ルパネル内で定義したフォーマットの DOUBLE	

AssetCenterの高度な使い方

# COMBOEDIT**コントロール**

このコントロールではドロップダウンリスト型のコントロールを挿入できます。

表 14.36. COMBOEDITコントロールのプロパティ

プロパティ名=値	プロパティの説明	例
VALUES	コントロールの値の組合せを定義します。	
	「名前」にはコントロールに表示するテキ	
	ストを定義します。「値」には、ユーザが	
	この「名前」を選択した場合コントロール	
	に割り当てる値を定義します。	
	文字列型のプロパティ	

## DATETIMEEDIT**コントロール**

このコントロールでは日付タイプのコントロールを挿入できます。

#### 表 14.37. コントロールのプロパティ

プロパティ名=値	プロパティの説明	例
FORMAT	コントロールのフォーマットを定義しま	2002/02/07
	す。	13:37:19
	• 日付(date)	2002/02/07 13:37:19
	• 時刻(time)	
	• 日付 + 時刻	
	コントロールのフォーマットは、ユーザが 定義するシステムパラメータにより変化し ます。	

# ウィザードの作成例

ウィザードのプログラミングを論理的に説明するために、「移動」ウィザード を作成します。「データベース」タイプのアクションと関連させることで、ユー ザと関連資産をある設置場所から他の設置場所へ移動するプロセスを簡略化し ます。このウィザードの作成方法の手順を次に説明します。この手順に従って 実際にウィザードを作成し、問題が発生した場合に参考にしてください。

- 1 手順1:ニーズを分析する
- 2 手順2:ウィザードの編成方法を定義する
- 3 手順3:スクリプト言語を使ってウィザードの構造を記述する

## 手順1:ニーズを分析する

このウィザードの目的は、資産をある設置場所から別の設置場所へ移動することです。このため、次の操作を行う必要があります。

- 1 移動する資産を指定する。
- 2 これらの資産の新しい設置場所を選択する。

#### 移動する資産の指定方法

移動する資産を指定するには、3つの方法があります。

- 資産のユーザから移動する資産を特定する。ユーザを選択した後に、移動する資産を選択する必要があります。
- [資産]テーブルのレコードを選択して、移動する資産を直接選択する。
- 設置場所から移動する資産を特定する。まず、設置場所を選択し、次にこの 設置場所から移動する資産を選択します。

💋 注意:

ユーザが、移動する資産の選択方法をこれらの中から選択できるよう、ユーザ 選択ページを作成する必要があります。

#### 新しい設置場所を選択する

資産の新しい設置場所を選択するには、設置場所のテーブルからレコードを選 択するだけです。

## 手順2:ウィザードの編成方法を定義する

手順1で定義したニーズに従って、次のように、ウィザードの編成を定義する必要があります。

- 1 ページ数
- 2 ページ同士のリンク方法
- 3 各ページの内容

#### 💋 注意:

手順1で、選択ページを作成する必要があることがわかりました。このページが ウィザードの最初のページになります。このページを「AssetSelectionType」と 呼ぶことにします。

次の図を参考にして、ウィザードの定義方法を定義します。

#### 図 14.2. ウィザードの編成



#### このフローチャートを使って、ページごとのトランジションを定義できます。

ページ	トランジションのリンク先ページ
AssetSelectionType	SelectAsset, SelectUser, SelectLocation
SelectAsset	SelectNewLocation
SelectUser	SelectAsset
SelectLocation	SelectAsset
SelectNewLocation	なし

次に、ページの内容を定義します。つまり、ユーザが選択を実行できるコント ロールを定義します。

ページ	このページの目的	使用するコントロール
AssetSelectionType	ユーザが3つの選択肢から選 択できるようにします。	CHOICEコントロール
SelectAsset	ユーザが <b>[資産]</b> テーブル のレコードリストから資産 を選択できるようにしま す。	ADBLISTコントロール
SelectUser	ユーザが、 <b>[部署と従業</b> 員]テーブルから資産を移 動するユーザを選択できる ようにします。	ADBLISTコントロール
SelectLocation	ユーザが、 <b>[場所]</b> テーブ ルから現在の設置場所を選 択できるようにします。	ADBLISTコントロール
SelectNewLocation	ユーザが、 <b>[場所]</b> テーブ ルから新しい設置場所を選 択できるようにします。	ADBLISTコントロール

# 手順3:スクリプト言語を使ってウィザードの構造を記述する

この手順では、ウィザードのスクリプトを作成します。スクリプトには、ウィ ザードの各ノードの構造を記述します。下に、「移動」ウィザードのコメント 付きソースノードを示します。このコードは、ウィザードの作成の一例にすぎ ません。同じタスクを実行するウィザードの作成方法は、他にもたくさんあり ます。

;=====================================
,====================================
,

```
ストの場合、amEmpIDeptから選択
_____
_____
{ PAGE pgUser
TITLE = "移動する従業員の選択"
{ DBLISTBOX Users
 COLNAME = "Name|FirstName"
 COLWIDTH = "50|50"
 DBLCLICK = 1
 LABEL = "移動する従業員"
 MULTISEL = 1
 TABLE = "amEmpIDept"
 { VALUE =
  if [CurrentTable] = "amEmpIDept" then
  RetVal = [CurrentSelection]
  else
   RetVal = ""
  end if
 }
 VISIBLE = 1
{ TRANSITION trPersonToNewLoc
 TO = "pgNewLoc"
}
                         _____
     _____
新しい設置場所を指定するページ
{ PAGE pgNewLoc
TITLE = "新しい設置場所の選択"
{ STRING UserName
 VALUE = AmDbGetString("SELECT FirstName + ' ' + Name FROM amEmpIDept
WHERE IEmpIDeptId IN (" & {pgUser.Users} & ")")
{ LABEL LABEL1
 CAPTION = " \neg - \vartheta: " & {UserName}
}
{ DBLISTBOX NewLocId
 COLNAME = "名前"
 COLWIDTH = "100"
```

```
DBLCLICK = 1
 TABLE = "amLocation"
 VALUE = "-1"
}
{ TRANSITION trNewLocToAssets
 TO = "pgRecap"
}
     _____
選択内容の確認ページ
  _____
PAGE pgRecap
TITLE = "Recapitulatif"
 { LISTBOX Users
 COLTITLE = "名前"
 COLWIDTH = "100"
 LABEL = "移動する従業員"
 MANDATORY = 0
 MULTISEL = 1
 READONLY = 1
 VALUE = ""
 VALUES = AmDbGetList("SELECT FullName FROM amEmpIDept WHERE FullN
ame LIKE LikeParam(amEmpIDept_2:FullName)+'%' AND amEmpIDept_2:IEmpIDep
tId IN(" & {pgUser.Users} & ")","|",",", "=")
}
     _____
最後のページ
_____
     ------
FINISH FINISH
{ DO =
 On Error Goto ErrHandler
 Dim IErr as long
 dim hRecord as Long
 dim iEmplCount as Integer
```

```
iEmplCount = {pgRecap.Users.VALUES.Count()}
 dim iMax as Long
 iMax = iEmplCount
 dim ILocald as long
 ILocald = {pgNewLoc.NewLocId}
 IErr = amStartTransaction()
 dim i as Integer
 For i = 1 To iEmplCount
  IErr = AmProgress((100 * i ) / iMax)
  IErr = AmLog("Deplacement de la personne " + {pgRecap.Users.VALUES(i,1)})
  hRecord = AmGetRecordFromMainId("amEmpIDept", {pgRecap.Users.VALUES(i
,0)})
  If hRecord <> 0 then
   IErr = AmSetFieldLongValue( hRecord, "ILocald", ILocald)
   IErr = AmUpdateRecord(hRecord)
   IErr = AmReleaseHandle(hRecord)
  End If
 Next i
 IErr = amCommit()
 RetVal = 0
 Exit Function
 ErrHandler:
  On Error Goto 0
  AmLog(AmLastError() & " - " & AmLastErrorMsg())
  AmLog("La transaction a ete annulee")
  RetVal = 1
  Exit function
SUMMARY = 1
```

# グラフィカルエディタの使用方法

AssetCenterでは、内蔵のグラフィカルエディタを使ってウィザードを作成でき ます。このエディタは、ウィザードの作成プロセスを簡略化および高速化する ことを目的としています。スクリプト言語の代わりに使用するものではありま せん。ウィザードを設計するには、スクリプト言語を理解できることが必要で す。

- インタフェースの概要
- 新しいノードを作成する
- ノードのプロパティを編集する
- ウィザードをコンパイル、実行、およびデバッグする

#### 💋 注意:

グラフィカルエディタを使用するには、作成または変更するアクションのタイ プ(SQL名:seActionType)が「ウィザード」のアクションでなければなりませ ん。

## インタフェースの概要

グラフィカルエディタにアクセスするには、[ツール/アクション/編集]メ ニューを選択します。グラフィカルエディタは、アクションの詳細画面の[ウィ ザード]タブページに表示されます。このエディタは次の3つの部分から構成さ れています。

- 最も一般的な機能を含むツールバー
- ウィザードの構造をツリー構造で示す[**階層**]セクション
- 階層で選択されたノードのプロパティの一覧を表示するセクション

#### ツールバー

ツールバーでは、編集コマンドを直接起動できます。アイコン上にしばらくマウスポインタを置くと、ツール名が表示されます。

#### 編集コマンド

4つの編集コマンドを使用できます。

- Aは、エディタのテキストモードとグラフィックモードを切り替えます。
- ▲は、同じ親ノード内の1つ上位のノードに移動します。
- ■は、同じ親ノード内の1つ下位のノードに移動します。
- 🔀 は、選択したノードを削除します。

#### 実行およびデバッグコマンド

これらのコマンドを使って、スクリプトをコンパイル、デバッグ、および実行 できます

図 14.3. 実行およびデバッグ ボタン

🜮 ×

#### 検索ツール

ツールバーには、ウィザードのツリー構造で文字列を検索できる検索ツールが あります([Ctrl+F]キーで直接このコントロールを起動できます)。 このゾーンをクリックし、検索するテキストを入力します。該当するテキスト が見つかった場合は、その場所が強調表示されます。([F3]および[Shift+F3] キーを使って、それぞれ次のまたは前の一致文字列に移動できます。)

#### 💋 注意:

テキストモードの場合は、テキスト全部を検索します。グラフィックモードの 場合は、プロパティ名だけで検索します。

#### ウィザードのツリー構造

グラフィカルエディタの左側には、ウィザードがツリー構造で表示されます。 ツリー内のノードを選択すると、画面の右側に、このノードに関連するプロパ ティが一覧表示されます。

## 選択したノードに対応するプロパティのリスト

画面の右側では、ノードのプロパティの値を入力できます。 各プロパティには固定値またはスクリプトがあります。プロパティの種類は次 のように色分けされています。

- プロパティでデフォルト値が使われている場合は、その名前と値がグレーで 表示されます。このプロパティに他の値を指定できます。その値は黒で表示 されます。
- プロパティでユーザ定義値またはスクリプトが使われている場合は、その名前と値が黒で表示されます。
- 必須プロパティの場合は、その名前と値が赤で表示されます。
- 変更された値は、青で表示されます。

## 新しいノードを作成する

ここでは、ノードで実行できる操作について説明します。ツールバーでは、ノードを上下に移動したり、削除したりすることができます。まず、新しいノードの作成について説明します。

💋 注意:

ポップアップメニューを使って、ノードを上下に移動したり、削除したりする こともできます。この場合は、選択したノードをマウスの右ボタンでクリック します。

ノードを作成するには、まず、親ノードを選択します。例えば、新しいPAGE ノードを作成するには、最初にROOTノードをクリックします。親ノードを選択 したら、マウスの右ボタンでクリックしてポップアップメニューを表示します。 [新規作成]メニューを選択すると、作成できるノードがグループごとに表示 されます。

ウィザードのツリーにノードが挿入されます。

#### ノードのプロパティを編集する

ノードを作成したら、ノードのプロパティに値を割り当てることができます。 この作業は、エディタの右側で行います。

プロパティの値は、次の2つの方法で定義できます。

- 固定値を入力する。
- スクリプトを定義する。

💋 注意:

スクリプトは常に、固定値より優先されます。プロパティにスクリプトと固定 値の両方を割り当てると、固定値は無視され、スクリプトが解釈されます。

#### プロパティに固定値を割り当てる

対象のプロパティの[値]列を直接クリックします。プロパティで使用可能な データ型(テキスト、ブール、倍精度数値など)に従って、リストから値を選 択するか、テキスト編集ゾーンに入力します。

#### プロパティにスクリプトを割り当てる

スクリプトを割り当てるプロパティを選択します。スクリプトそのものは、プロパティのリストの下にある**[スクリプト]**フィールドに入力します。

#### 💋 注意:

ポップアップメニュー(プロパティをマウスの右ボタンでクリックして表示) から[デフォルト値に戻す]を選択すると、固定値またはスクリプトが取り消 され、デフォルト値に戻ります。この操作は、値またはスクリプトがユーザに よって定義されたプロパティ(黒で表示される)でのみ使用可能です。

# ウィザードをコンパイル、実行、およびデバッグする

エディタのツールバーの<sup>98</sup> ボタンをクリックして、ウィザードを起動できま す。実行中に発生したエラーは、エラー履歴ウィンドウ(内蔵のウィザードデ バッガからアクセス可能)に表示されます。[Shift+F9]キーを押すと、実行を 中断し(ウィザードが固定ウィンドウの場合)、デバッガを起動できます。 このようにして、ウィザードのエラーを簡単に訂正できます。

💋 注意:

ウィザードがコンテキストに依存する場合は、実行ボタンを使用できません。

# ウィザードに関する一般的な質問

ここでは、ウィザードの作成時に起こり得る問題の対処法を紹介します。



次のコードが機能しません。

{lbxMyListBox.Values.Count}

## 答

メソッドのシンタックスに開き括弧と閉じ括弧を入力する必要があります。正 しいコードは次のようになります。

{IbxMyListBox.Values.Count()}

## 質問

次のコードが機能しません。

{IbxMyListBox.Line(IRow)}

## 答

LINEメソッドは、LISTBOXコントロールのVALUESプロパティに関連付けられ ます。正しいコードは次のようになります。

{IbxMyListBox.Values.Line(IRow)}

# 質問

次のコードが機能しません。

{lbxMyListBox.Values.Line({lbxTmp})}



メソッドでは参照するプロパティを使用できません。次のように記述します。

Dim IRow As Long IRow = {IbxTmp} {IbxMyListBox.Values.Line(IRow)}

### 質問

プロパティに固定値を割り当てる次のコードが機能しません。

{Property} = 123

## 答

プロパティに値を割り当てるには、次のように、AmSetProperty()関数を使う必要があります。

Dim irc as Integer irc= AmSetProperty("Property", 123)

AssetCenterの高度な使い方

#### 💋 注意:

戻りコードを使う必要がない場合でも、必ず戻りコードを取得するようにして ください(この場合は「irc」)。

#### 質問

データベースに資産を作成するウィザードを実行すると、次のエラーメッセージが表示されます。

12001 - 書き込み権限がありません。

ユーザが管理者として接続している場合も、このメッセージが表示されます。

## 答

このメッセージは、ウィザードのFINISH.DOノード外で書き込みアクセスを試みると表示されます。ウィザードは次の操作を実行します。

- 1 AssetCenter管理者も書き込みできない一連のページから情報を収集する。
- ユーザの権限に応じて書込みできるFINISH.DOノードに含まれるスクリプト を実行する。

#### 質問

ウィザードの実行中に表示されるエラーメッセージが不完全なことがあります。

#### 答

[Shift+F9]キーを押して、デバッグを表示します。多くの場合、履歴ウィンド ウのエラーメッセージの方が明確です。

## 質問

ウィザードのページでDBLISTBOXコントロールを使うと、パフォーマンスが低下します。これは普通のことなのでしょうか。



この問題は、フィルタと一緒にDBLISTBOXコントロールを使うと発生します。 この場合、選択を変更するたびに、データベースにクエリが送られ、選択がフィ ルタに一致するかどうかが確認されます。選択がユーザによって設定された場 合、この追加クエリは実行されません。

#### 質問

LISTBOXコントロールで特定の列を編集可能または編集不可にするにはどうすればよいですか。

答

このコントロールのEDITABLEプロパティを使います。このプロパティに割り 当てられる値は「0」と列の区切り文字として使われるパイプ文字())で区切ら れる文字で構成されています。「0」は列を「編集不可」として、「1」は列を 「編集可能」として定義します。値を省略すると、対応する列は編集できず、 列2と4が編集可能になります。

EDITABLE = "|1||1"



ウィザードで詳細ウィンドウを開くようにするにはどうすればよいですか。



ウィザード内でDDEコール(関数から)を使う必要があります。ウィザードは 固定ウィンドウにしないでください。ウィザードから資産テーブルを開く方法 は、以下の通りです。

Dim irc as Long irc = AmActionDDE("am", "AssetCenter", "OpenTable(amAsset)")

督問

LISTBOXコントロールの「COLNAME」プロパティと「COLTITLE」プロパティ の違いを教えてください。



LISTBOXコントロールの列のタイトルは、自動的に、または手動で定義できます。

- 「TABLE」プロパティに関連付けられた「COLNAME」プロパティでは、 データベースのフィールドラベルを使って、LISTBOXコントロールの列タイ トルを自動的に定義できます。
- 「COLTITLE」プロパティに値が入力されている場合は、その値が列のタイトルとして使用されます。このプロパティが定義されていない場合は、列タイトルは「COLNAME」プロパティで定義されたタイトルになります。

例

... TABLE = "amEmpIDept" COLNAME = "Name||FirstName" COLTITLE = "|A|B"

この例の場合、LISTBOXコントロールには、Name、A、Bというラベルが表示 されます。

「COLNAME」プロパティでは、列値が編集可能な場合に、使用するコントロールのタイプも定義します。

本章では、AssetCenterのユーザにニュースを配信し、管理する方法について説 明します。

ニュースのリストを表示するには、**[ツール/ニュース]**メニューを選択します。

ニュースを表示するには、**[ウィンドウ/ニュースマーキーの表示]**メニュー またはツールバーの <sup>Sel</sup> アイコンを使います。

# ニュースの概要

1

ニュースとは特定の期間に指定した受信者のグループに流す新情報のことを指 します。

これらの従業員は従業員グループに所属します。

原則として、ニュースは短い期間に関する情報です。

例:「2002/02/10、11:00と12:00の間にサーバXXXがダウンします。」

## ニュースの概要

# 新しいニュースを作成する

ニュースのテーブルでのニュース作成の権限を持つAssetCenterのユーザは、 [ツール/ニュース]メニューを使ってニュースを作成できます。 ニュースの詳細には、主に次の情報が表示されます。

- メッセージ
- メッセージの配信先リスト
- メッセージの有効期間

### ニュースを表示する

ニュースの配信先リストに含まれているユーザは、ニュースマーキーを表示で きます。

## ニュースの重要度

ニュースメッセージの重要度を定義するには、ニュースの詳細画面にある[重 要度](SQL名:seSeverity)フィールドで指定します。 各重要度には[色](SQL名:IColor)フィールドで選択された色が関連付けら れます。ニュースマーキーには、この色でニュースが表示されます。

## 配信するメッセージ

ニュースの詳細画面の**[メッセージ]**タブページには、配布するメッセージが 表示されます。

メッセージの長さは最大255文字です。すべての文字を使用できます。

# ニュースの配信先リスト

ニュースの詳細画面の[**配信先**]タブページには、そのニュースを読むことが できる従業員グループの一覧が表示されます。

# [すべての従業員グループ] (SQL名:bAllGroups) チェックボックス

このチェックボックスをオンにすると、すべての従業員グループのメンバーは AssetCenterでメッセージを表示できます。

チェックボックスがオフの場合は、**[配信先]**タブページで選択された従業員 グループのメンバーのみがメッセージを表示できます。

配信先リストの従業員グループを追加、削除、表示、または変更するには、それぞれ <br />
■、および <br />
■ ボタンを使います。

## [サブグループを含む] (SQL名:bChildGrps)チェッ クボックス

ニュースのメッセージは、全従業員グループまたは選択したグループだけに配 信できます。従業員グループのテーブルは階層構造になっているため、この チェックボックスをオンにすると(デフォルトではオン)、選択されたグルー プの全サブグループもニュースを表示できます。

# ニュースを表示する

ニュースを表示するには、次の操作を行います。

- 1 ニュースマーキーを起動します。
- 2 マーキーの右側にあるボタンを使って、複数のニュースをスクロールおよび 参照します。

次の内容を定義できます。

- ニュースの重要度に応じたマーキーの配色
- ニュースマーキーツールバーの自動更新モード

### ニュースマーキーを起動する

AssetCenterのすべてのユーザがニュースマーキーを表示できます。現在ログインしているユーザに関連するメッセージが表示されます。ニュースマーキーは、次の2つの方法で起動および終了できます。

- [ウィンドウ/ニュースマーキーの表示]メニューを使う。
- ツールバーの 🕯 アイコンを使う。

#### 表 15.1. マーキーツールバーのボタン

<b>a</b>	オプションで定義されている時間を待たずに( <b>[編集/</b> <b>オプション]</b> メニュー )、即座に現在のニュースを読む
	ことができます。
•	前のメッセージを表示します。
	次のメッセージを表示します。
11	表示されているメッセージを中断または再開します。

