

データベースアクセスコンポーネント

1. 概要

データベースアクセスコンポーネントとは、SQL データベースにアクセスして SQL 文を実行することによりデータベース検索を行う機能を提供するコンポーネントです。また、データベースアクセスコンポーネントでは、データベースの構成情報、接続情報、エラー情報等を取得することも可能です。データベースアクセスコンポーネントは、アプリケーションビルダーのメニューから以下のように選びます。

[コンポーネント追加] - [入出力] - [データベース] - [データベースアクセス]

このコンポーネントの使用には、JDBC を介してアクセス可能な MySQL 等の SQL データベースが必要です。MySQL のインストールに関しては「工程管理システム導入マニュアル」を参照してください。

2. 用途

- ・ データベース検索を行いたいとき。
- ・ Microsoft Access や Excel のテーブルデータを直接的に操作したいとき。

3. ここで使用されるイベントとメソッド

データベースアクセスコンポーネントから発生するイベントを次に示します。

- ・ データ生成イベント

表 1 データベースアクセスコンポーネントから発生するイベント（データ生成イベント）

トリガ	対象データ	イベント番号
「SQL 文を実行する (String)」を使用し、データを返す SQL 文を実行したとき	検索結果（テーブルデータ）	0

「イベント番号を指定して SQL 文を実行する (String, int)」を使用し、データを返す SQL 文を実行したとき	検索結果 (テーブルデータ)	指定したイベント番号
「テーブルに記述した SQL 文を実行する (PFObjectTable)」を使用し、データを返す SQL 文を実行したとき	検索結果 (テーブルデータ)	テーブルの第 2 列に指定されたイベント番号

- ・ 処理完了イベント

表 2 データベースアクセスコンポーネントから発生するイベント (処理完了イベント)

トリガ	処理結果データ	イベント番号
「SQL 文を実行する (String)」を使用し、データを返さない SQL 文を実行したとき	更新された行数 (レコード数)	0
「イベント番号を指定して SQL 文を実行する (String, int)」を使用し、データを返さない SQL 文を実行したとき	更新された行数 (レコード数)	指定したイベント番号
「テーブルに記述した SQL 文を実行する (PFObjectTable)」を使用し、データを返さない SQL 文を実行したとき	更新された行数 (レコード数)	テーブルの第 2 列に指定されたイベント番号
データベースに接続、切断したとき	0	0
SQLException が発生したとき	0	-1

データベースアクセスコンポーネントの主なメソッドは次のとおりです。

表 3 データベースアクセスコンポーネントの主なメソッド

メソッド名	機能説明
データベースに接続する (String, String, String, String)	引数にそれぞれドライバ名、データベースの URL、ユーザ名、パスワードを指定し、データベースに接続する。
データベースとの接続を切断する ()	接続中のデータベースとの接続を切断する。
SQL 文を実行する (String)	引数として SQL 文を指定し、データベース検索を実行する。検索結果は発生したイベントから取得するか、戻り値として取得することができます。

イベント番号を指定して SQL 文を実行する (String, int)	第 1 引数に SQL 文、第 2 引数にこのコンポーネントから発生するデータ生成イベントまたは処理完了イベントのイベント番号を指定して、データベース検索を実行する。
テーブルに記述した SQL 文を実行する (PFObjectTable)	引数に SQL 文を記載したテーブルを指定して、データベース検索を実行する。
接続情報、エラー情報等を取得する ()	接続情報、エラー情報等を文字列として取得する。
getDatabaseTree () (日本語未設定)	接続中のデータベースの構造をツリーデータとして取得する。
setDatabase (String) (日本語未設定)	引数にデータベース名を設定して、使用するデータベースを指定する。

メソッドの機能説明について補足します。

- ・データベースに接続する

第 1 引数のドライバ名には、使用するドライバ名を指定します。例えば MySQL に JDBC ドライバを用いて接続する場合には、“com.mysql.jdbc.Driver” と記入します。

第 2 引数のデータベースの URL には、データベースの「サーバ名」+「データベース名」を指定します。データベース、ドライバによって異なりますが、MySQL と上述のドライバを用いる場合には、“jdbc:mysql://ホスト名/データベース名” と記入します。MZ Platform と同じパソコンに MySQL がインストールされている場合、ホスト名は“localhost” と記入します。また、データベースの種類や設定によっては、検索結果に日本語が含まれる場合、文字化けする場合があります。この場合、使用する日本語コードを“jdbc:mysql://ホスト名/データベース名?useUnicode=true&characterEncoding=SJIS” というように明示することにより、解決する場合があります。詳細は各データベースの解説書をご覧ください。

第 3 引数のユーザ名には、データベースに登録されているユーザ ID を指定します。

第 4 引数のパスワードには、上述のユーザ ID に対して設定されているパスワードを指定します。

- ・SQL 文を実行する

引数の SQL 文として“SELECT”文等のデータを返す SQL コマンドを実行した場合、データ生成イベントが発生し、検索結果はイベントに内包されたテーブルデータとして取得することができます。また戻り値としても取得することができます。

一方、“INSERT”、“UPDATE”、“DELETE”等何も返さない SQL コマンドを実行した場合、処理完了イベントが発生し、変更された行数（レコード数）がイベントの内包データ（整数）として取得できます。

ただし、SQL コマンドは、データベースの種類によって異なりますので、詳細は市販の参考書をご覧ください。

- ・イベント番号を指定して SQL 文を実行する

上述のメソッド「SQL 文を実行する」を、発生するイベントのイベント番号を指定して実行することができます。発生するイベントの種類は、上述のメ

ソッドに準じます。イベント番号の設定以外、操作はメソッド「SQL 文を実行する」と同様ですので、説明を省略します。

- テーブルに記述した SQL 文を実行する

引数の「テーブルデータ」は N 行 2 列とし、第 1 列には SQL 文、第 2 列には第 1 列の SQL 文が実行後に発生するイベントのイベント番号を記載します。第 1 行から順に第 N 行まで SQL 文が実行され、発生するイベントの種類は上述のメソッドと同じです。SQL 文をテーブルとして設定する以外、操作はメソッド「SQL 文を実行する」と同様ですので、説明を省略します。

- 接続情報、エラー情報等を取得する

データベースへのアクセス時に発生する情報を文字列として取得することができます。データベースの接続状態やエラー等が取得できます。

- getDatabaseTree

接続中のデータベースの構造がツリーデータとして取得できます。ルートはデータベース製品名、ノードはデータベース名、リーフはテーブル名となります。

- setDatabase

引数に使用するデータベース名を指定することにより、検索対象のデータベースを変更することができます。接続時に指定したデータベースから使用するデータベースを変更する場合などに使用します。

4. コンポーネント使用例

データベースアクセスコンポーネントを用いたサンプルアプリケーションが AD_DATA¥Sample に収録されています。このサンプルアプリケーションは、サンプルのデータベース「production」が既に存在することを仮定しています（他のデータベース、テーブルでも構いません）。サンプルのデータベース「production」に関しては、「工程管理システム操作マニュアル」または「工程管理システム導入マニュアル」を参照してください。

ビルダーまたはローダーから、「データベースアクセス_1.mzax」をロードし実行してください。図 1 のようにフレームが 2 つ表示されます。

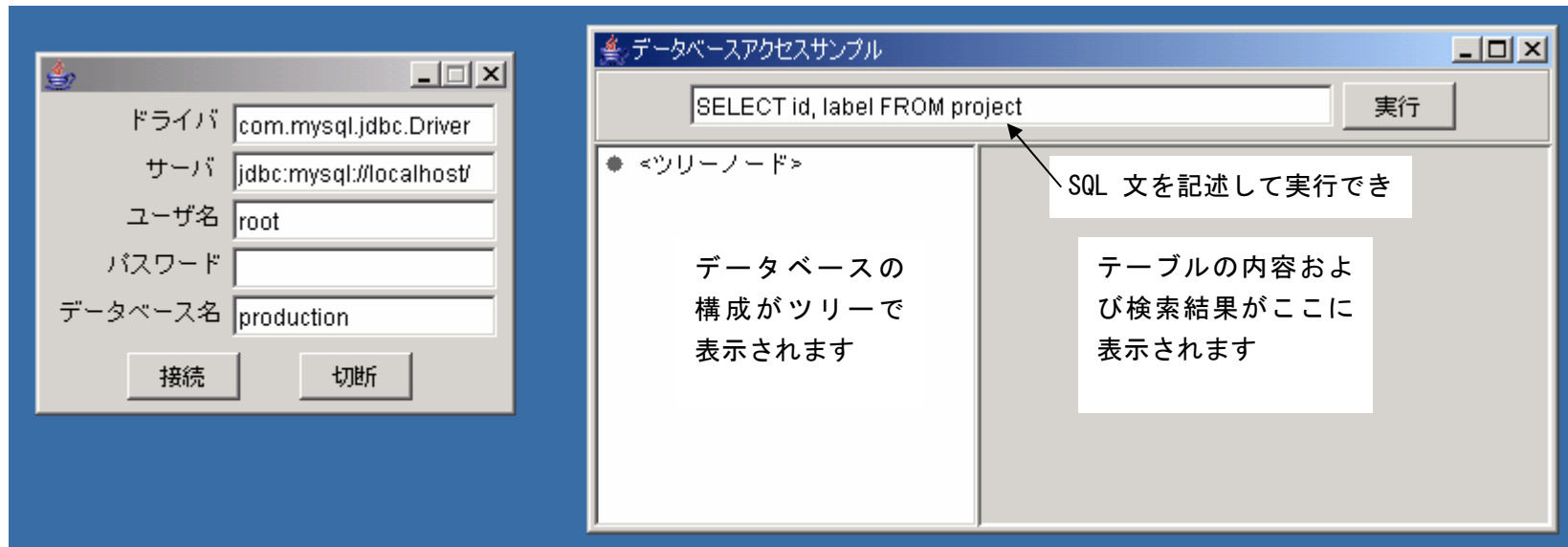


図 1 データベースアクセスサンプルアプリケーション

左側のフレームには接続するデータベースシステムのドライバ名、サーバ名、ユーザ名、パスワード、データベース名を設定します。これらが正しく設定されている場合には、[接続]ボタンを押すことによりデータベースに接続できます。また[切断]を押した場合には接続中のデータベースから切断できます。右側のフレームには、データベースの構成、テーブルの内容が表示されます。テキストフィールドに SQL 文を記述して[実行]ボタンを押すことにより、テーブルを検索することが可能です。

注意：データベースの種類や設定によっては、検索結果に日本語が含まれる場合、文字化けする場合があります。この場合、使用する文字コードを指定することにより解決する場合があります。MySQL の場合には、図 1 の左側のフレームのデータベース名の欄に、“データベース名?useUnicode=true&characterEncoding=SJIS”と記述し、再接続します。詳細は各データベースの解説書をご覧ください。

データベースに接続すると、右のフレームは図 2 のようになります。



図 2 データベース接続時のイメージ

フレーム中のツリーにはデータベースの構造が表示されます。ルートにはデータベース製品名と URL が、ノードにはデータベース名が表示されます。ノードを開くと、図 3 に示すようにデータベースを構成するテーブル名が表示され、テーブルを選択すると内容を閲覧することができます。

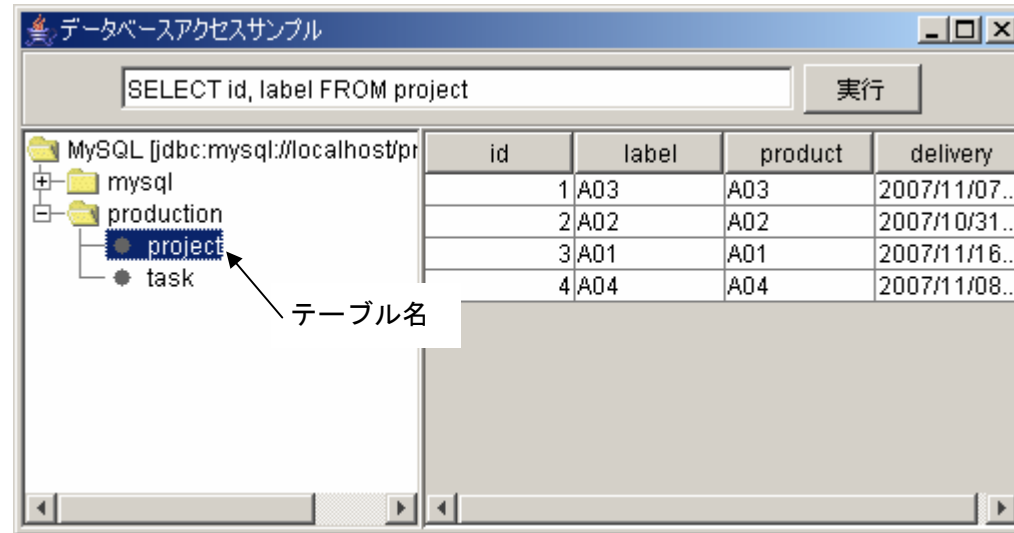


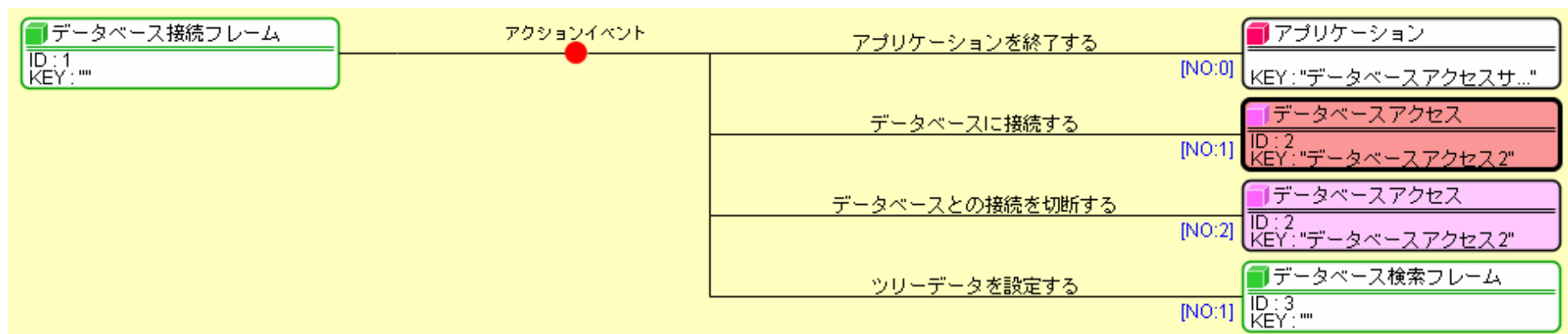
図 3 テーブル内容参照時のイメージ

フレーム上部のテキストフィールドに SQL 文を記述し、[実行]ボタンを押すと SQL 文による検索結果を表示することができます。あらかじめ記述されている「SELECT id, label FROM project」はテーブル「project」からフィールド「id」と「label」だけで構成される部分を選択し表示する SQL 文です。この SQL 文を実行すると図 4 のように検索結果が表示されます。SQL 文に関する詳細は市販の参考書をご覧ください。



図 4 SQL 文による検索時のイメージ

次にサンプルアプリケーションでのコンポーネント間の接続について説明します。はじめに、「データベース接続フレーム」に記述された情報を使用してデータベースに対する接続及び切断を行うため、次のようにコンポーネント間を接続しています。



M2 起動メソッド情報

メソッド データベースに接続する (String,String,String,String) 全メソッド対象

NO	型	説明	取得方法	コンポーネント	メソッド/値
0	String	ドライバ名	メソッド戻り値	データベース接続フレーム [ID:1] (KEY:"")	ドライバ名を取得する
1	String	データベースのURL	メソッド戻り値	データベース接続フレーム [ID:1] (KEY:"")	データベースのURLを取得する
2	String	ユーザ名	メソッド戻り値	データベース接続フレーム [ID:1] (KEY:"")	ユーザ名を取得する
3	String	パスワード	メソッド戻り値	データベース接続フレーム [ID:1] (KEY:"")	パスワード文字列を取得する

閉じる

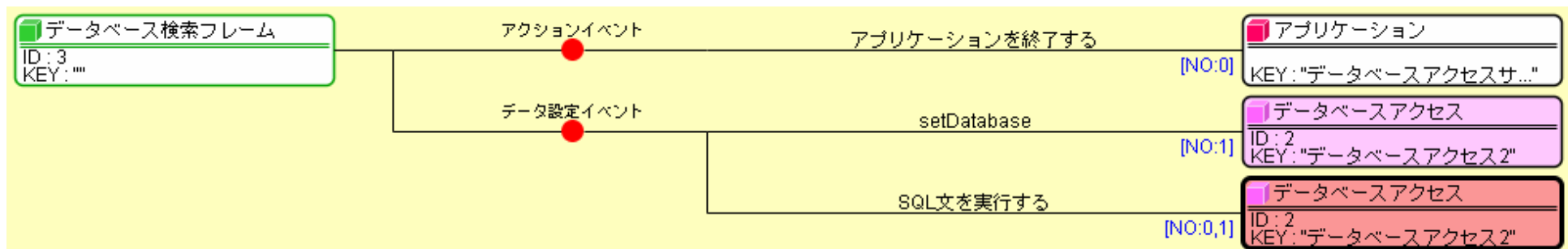
M2 起動メソッド情報

メソッド データベースとの接続を切断する() 全メソッド対象

NO	型	説明	取得方法	コンポーネント	メソッド/値
----	---	----	------	---------	--------

閉じる

データベースの検索に関しては、「データベース検索フレーム」から発生するデータ設定イベントに内包される SQL 文を取得して検索を実行するように、次のようにメソッドを使用しています。



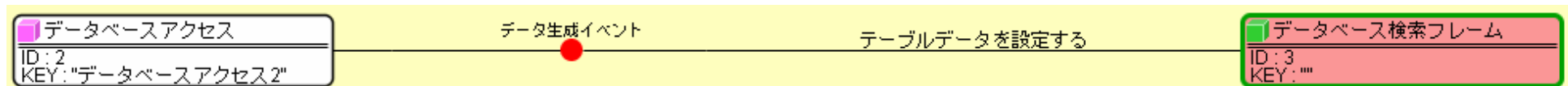
M2 起動メソッド情報

メソッド 全メソッド対象

NO	型	説明	取得方法	コンポーネント	メソッド/値
0	String	SQL文	イベント内包	-	イベント対象データ

閉じる

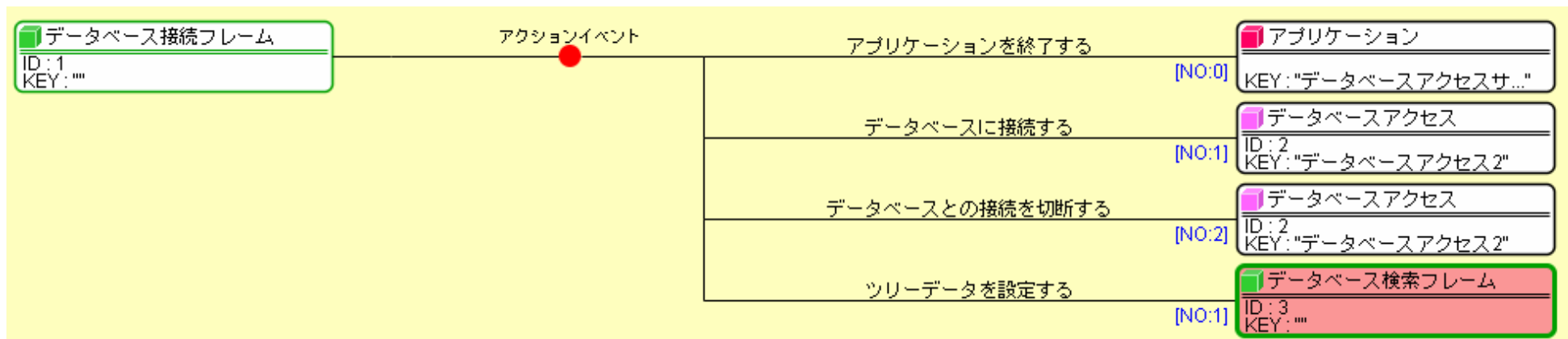
また、検索結果はデータベースアクセスコンポーネントから発生するデータ生成イベントにテーブルデータとして内包されており、そのテーブルデータを「データベース検索フレーム」のテーブルに設定しています。



M2 起動メソッド情報					
メソッド テーブルデータを設定する(PFObjectTable)					
NO	型	説明	取得方法	コンポーネント	メソッド/値
0	PFObjectTable	テーブルデータ	イベント内包	-	イベント対象データ

閉じる

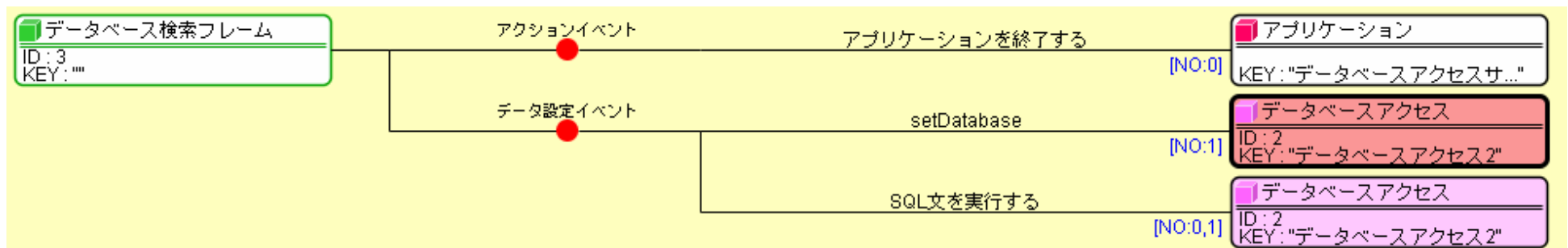
サンプルアプリケーションではデータベース接続時に、データベースの構造をツリーとして表示するように次の図のようにメソッドを使用しています。

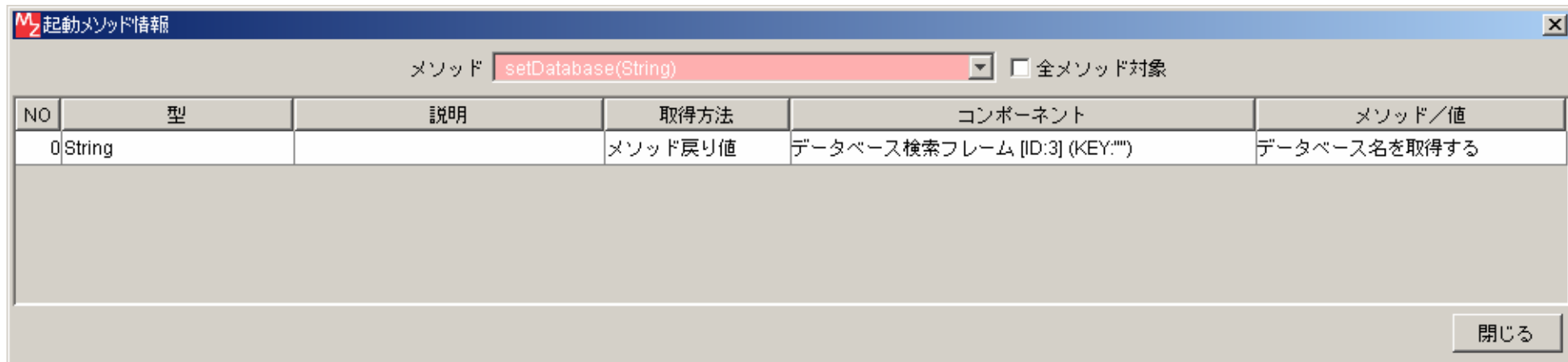


M2 起動メソッド情報					
メソッド ツリーデータを設定する (PFOBJECTTREE)					
NO	型	説明	取得方法	コンポーネント	メソッド/値
0	PFOBJECTTREE	ツリーデータ	メソッド戻り値	データベースアクセス [ID:2] (KEY:"データベースアク...	getDatabaseTree

閉じる

サンプルアプリケーションではツリーのテーブル名が選択された際に、使用するデータベースをそのテーブルが含まれているデータベースに切り替えるため、次の図のようにメソッドを使用しています。





参考のため、さらに機能を拡張したサンプルアプリケーションも AD_DATA¥Sample に収録されています。

はじめの拡張例は「データベースアクセス_1.mzax」を Microsoft Excel（エクセル）に接続するように変更を行ったものです。

ビルダーまたはローダーから「データベースアクセス_2(エクセル接続).mzax」をロードし実行してください。図 5 のようにフレームが 2 つ表示されます。エクセルファイルへの接続にはユーザ名及びパスワードを設定する必要がないため、その入力欄をフレームから削除してあります。ドライバには「sun.jdbc.odbc.JdbcOdbcDriver」、サーバ名には「jdbc:odbc:DRIVER=Microsoft Excel Driver (*.xls)」、データベース名にはエクセルファイルまでのパスを設定します。ファイルの選択は[参照...]ボタンを押してファイル選択ダイアログから行うこともできます。これらが正しく設定されている場合には、[接続]ボタンを押すことによりエクセルファイルに接続できます。また[切断]を押した場合には接続中のエクセルファイルから切断できます。右側のフレームには、ワークシートの構成及び内容が表示されます。ワークシート内のデータの検索するための SQL 文は、ワークシート名（テーブル名）は後ろに\$をつけダブルクォーテーションで囲むことが他のデータベースとは異なりますので、ご注意ください。

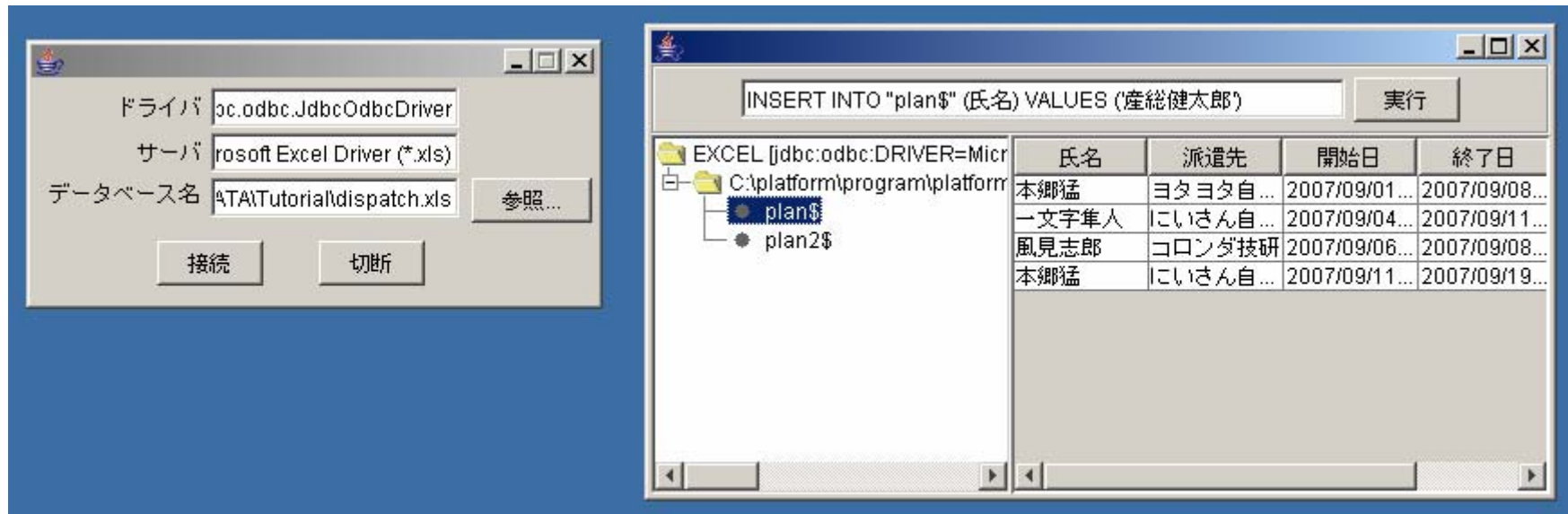


図 5 データベースアクセスサンプルのエクセル接続への変更

二つ目の拡張例は、テーブルのレコードの操作を行う機能を追加したものです。本サンプルアプリケーションは、MySQL、Postgres、JavaDB で動作を確認済みですが、他のデータベースに関しては修正が必要になる場合があります。また、Microsoft Excel 及び Access には未対応です。

ビルダーまたはローダーから、「データベースアクセス_3(機能拡張版).mzax」をロードし実行してください。図 6 のようにフレームが 2 つ表示されます。左側のデータベースへの接続を行うフレームでは、スキーマ対応のデータベース (Postgres 等) でスキーマを指定できるように拡張されています (MySQL ではバージョン 4. x. x 以前では使用しません)。一方、右側のフレームには、テーブルを作成及び削除するボタン、テーブルのレコードを操作するボタンが追加されています。「テーブル作成・削除」ボタンを押すと図 7 のようなテーブル作成・削除画面が表示されます。ここにテーブルを追加するデータベース名 (またはスキーマ名)、追加するテーブル名、各列の名前とタイプ、サイズ (VARCHAR 等で指定)、NULL 可否、主キーの指定等を指定して、「作成」ボタンを押すとテーブルが追加できます。一方、「削除」ボタンを押すと指定されたテーブルが削除されます。作成するテーブルには、主キーを必ず指定するよう注意してください。このサンプルアプリケーションでは、主キーが設定されていないと、レコードの更新と削除ができないようになっています。

右側の「参照」、「挿入」、「更新」、「削除」ボタンを押すと図 8 のように、ボタンに対応した操作画面が表示されます。このとき、テーブルのレコードを選択しておくこと、内容が自動的にフィールドに表示されます。各欄に記入または更新し、ボタンを押すと各処理が実行されます。このとき、主キーが設定されていないテーブルについては、更新及び削除処理が行えないのでご注意ください。

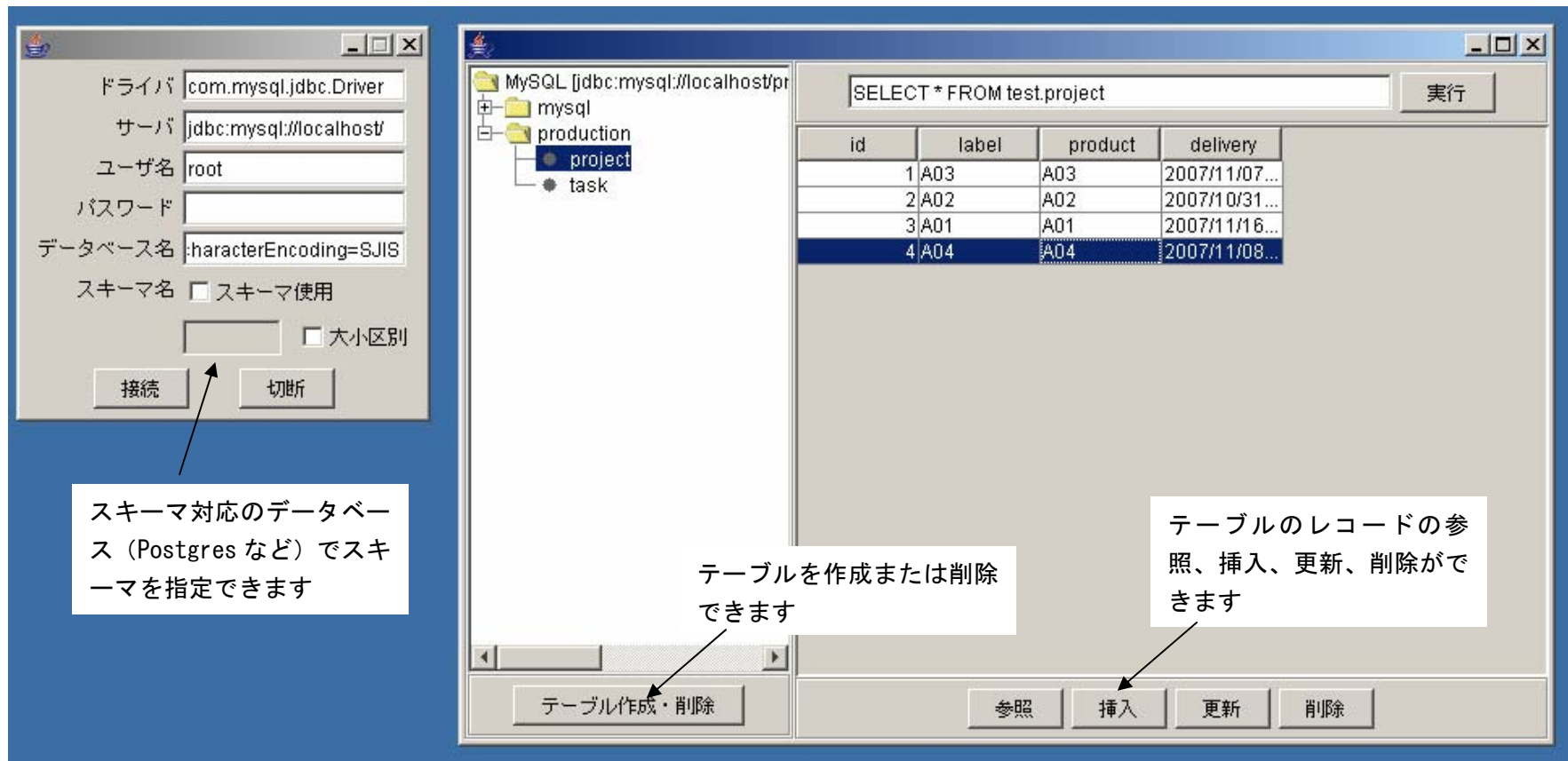


図 6 データベースアクセスサンプル拡張版



図7 テーブル作成画面



図8 レコード操作画面