



新機能ガイド

■ ごあいさつ

このたびは弊社製品 DataAnyware 2.0 をご購入いただきまして、誠にありがとうございます。

本書内で、各製品を以下名称で記述してご説明して参ります。

DataAnyware 2.0	本製品
DataAnyware 2.0 Server	Server
DataAnyware 2.0 Studio	Studio

本書は、追加された新機能、及び変更点についてご説明しています。

ご使用の際にお読みください。

なお、表示画面などは操作の一例として掲載しているものです。お客様のご使用環境によっては、画面に表示される内容が異なる場合がありますので、ご了承ください。

■ 著作権／ご注意

- 本書に記載されている会社名、製品名は、それぞれ各社の商標または登録商標です。
- 本書の内容の一部または全部を無断で複写転載することを禁じます。
- 本書に掲載の内容および製品の仕様などは、予告なく変更されることがあります。

本書の内容は万全を期して作成しておりますが、万一ご不明な点や誤り、記載もれ、乱丁、落丁などお気づきの点がございましたら、弊社までご連絡ください。

目次

■	ごあいさつ	2
■	版權／ご注意	2
1.	新機能ガイドについて	8
2.	マーク説明	9
2.1.	このマニュアルで使用しているマークについて	9
3.	全般	10
3.1.	種別	10
3.1.1.	種別の設定方法	10
3.1.2.	種別の使用方法	11
3.2.	HTTPS 通信	13
3.2.1.	設定方法	14
4.	インストーラ	16
4.1.	Java のバージョンの変更	16
5.	CLI Console.....	17
5.1.	クライアント CLI.....	17
5.2.	CLI Console ログ.....	17
5.3.	新規コマンド	18
5.4.	追加オプション	19
6.	DataAnyawre 2.0 Server	19
6.1.	RMI 通信ポートの固定化機能	19
6.2.	XML ログへのバージョン情報の出力	20
7.	ScriptRunner	21
7.1.	起動設定ファイルのパスワード暗号化	21
8.	Studio.....	22
8.1.	マイサービス	22
8.2.	マイプロジェクト.....	23
8.2.1.	プロジェクトアップロード時のデフォルト名付与	23
8.3.	デザイナー	24
8.3.1.	Undo/Redo	24

8.3.2.	プロパティウィンドウのサイズ変更	25
8.3.3.	プロパティウィンドウのサイズ自動拡張機能	25
8.3.4.	サービス名の指定	25
8.3.5.	プロジェクト/スクリプト仕様書の生成	26
8.3.6.	スレッド処理	27
8.3.7.	待機処理のミリ秒指定	27
9.	コントロールパネル	28
9.1.	アプリケーションログ出力先設定	28
9.2.	バージョン情報	29
9.3.	環境変数管理	29
9.4.	休日設定	30
9.5.	「DataAnywareServer 2.0 の設定」パッチ情報タブ	30
9.6.	「マウント設定」マウント解除時のオプション	31
9.7.	「マウント設定」マウントの変更	32
9.8.	「グローバルリソースの設定」グローバルリソースの ACL 機能	32
10.	マイログ	33
10.1.	プロジェクト名やスクリプト名での検索機能	33
10.2.	ログのダウンロード機能	34
11.	エクスプローラー	35
11.1.	テキストエディタで開くことのできるファイルサイズの指定	35
12.	ヘルプ	36
12.1.	ブラウザヘルプの対応	36
12.2.	オンラインヘルプ「同期ボタンの追加」	37
12.3.	新規追加の項目	37
13.	DB トリガー	38
13.1.	DB トリガーとは	38
13.2.	DB トリガーの使用方法	39
13.3.	DB トリガーの仕様	39
14.	Mapper 全般	40
14.1.	内部演算の仕様変更	40
14.2.	「条件」-「数値」カテゴリの Mapper ロジックの仕様変更	41
14.3.	Mapper ロジックの編集ダイアログ	42

14.4.	プロパティの追加.....	43
14.5.	null オプション.....	43
14.5.1.	オプションプロパティの使用法.....	44
14.6.	指定桁数オプション	44
14.6.1.	オプションプロパティの使用法.....	45
14.7.	トリムオプション.....	45
14.7.1.	オプションプロパティの使用法.....	45
14.8.	有効桁数の指定と丸め処理	45
14.8.1.	オプションプロパティの使用法.....	46
15.	Mapper	47
15.1.	新規追加された Mapper ロジック	47
16.	アダプタ全般	47
16.1.	パスワードの暗号化	47
17.	データベースアダプタ	48
17.1.	テーブル書き込み処理および更新系 SQL 実行処理のエラー時の詳細なダンプファイルの出力機能	48
18.	DB2 アダプタ	50
18.1.	DB2 V9.5 アダプタの追加	50
19.	Oracle アダプタ.....	50
19.1.	Oracle 11g アダプタの追加	50
20.	SQL Server アダプタ.....	51
20.1.	ローダによる書き込み処理の追加	51
20.2.	Windows 統合認証の対応	52
21.	SQL ウィザード.....	53
21.1.	SQL ステートメント再編集機能.....	53
22.	Excel アダプタ	54
22.1.	Excel2007 対応	54
22.2.	数式セルの値取得に関する仕様変更	54
22.3.	書き込み処理	54
23.	CSV アダプタ/Excel アダプタ	55
23.1.	列一覧の読み込み.....	55
24.	固定長アダプタ.....	56

24.1.	フィールドごとのエンコードの指定機能	56
25.	固定長/可変長ウィザード共通	58
25.1.	デザイナーでのフォーマット定義編集時の保存タイミングの変更	58
26.	Lotus Domino アダプタ	59
26.1.	Lotus Domino 7.0 アダプタの追加	59
26.2.	添付ファイル名の自動変換	59
27.	FTP アダプタ	60
27.1.	固定スキーマの自動設定	60
27.2.	コンポーネント変数の追加	60
27.3.	STFP、FTP over SSL 対応	61
27.3.1.	SFTP 接続	61
27.3.2.	FTP over SSL 接続	62
28.	メールアダプタ	63
28.1.	認証テスト機能	63
28.2.	差出人のメールアドレスのみを取得する（表示名は取得しない）機能	64
28.3.	添付ファイル名の自動変換	64
28.4.	新着順に受信する機能	64
28.5.	最大受信件数の指定機能	65
28.6.	受信できないメールをスキップする機能	65
28.7.	送信できない入力データをスキップする機能	66
28.8.	UID の取得機能	67
29.	リポジトリ DB	67
29.1.	リポジトリ DB とは	67
29.2.	リポジトリ DB の準備	67
29.2.1.	Oracle Database を使用する場合	68
29.2.2.	DB2 を使用する場合	69
29.2.3.	Microsoft SQL Server を使用する場合	69
29.2.4.	PostgreSQL を使用する場合	70
29.3.	リポジトリ DB の適用設定	70
29.3.1.	Oracle Database を使用する場合	71
29.3.2.	DB2 を使用する場合	73
29.3.3.	Microsoft SQL Server を使用する場合	75

29.3.4. PostgreSQL を使用する場合77

1. 新機能ガイドについて

本ドキュメントは Version 1.1.0 と比較しての新機能・変更点をまとめております。
新規に追加された機能に関しては、「NEW」アイコンを付加して記述してあります。

2. マーク説明

2.1. このマニュアルで使用しているマークについて

本マニュアルでは、以下のマークを利用して説明を進めていきます。

	操作手順の始まりを示します。
	本文中で説明している内容に関連する説明がコラムとしてまとめられています。必要なときにお読みください。
	機能や操作手順に対するメモを示します。
	操作上の注意や機能の制限事項を示します。必ずお読みください。
	前リリースバージョンと比較して、新規に追加された機能であることを示します。
	機能や操作手順に対する補足的な説明を示します。

また、説明は次の規則に沿って行われています。

- 画面に表示されるメニュー名、タブ名、プロパティ項目名および値、ボタン名は[]で囲んで表します。また、それ以外の機能名やプロパティウィンドウのタイトル、名称のないものは「」で囲んで前者と区別しています。
- `${dataanyware.home}`は本製品をインストールしたディレクトリを表します。デフォルトでは、「C:¥Program Files¥DataAnyware」となります。
- “<”と“>”で囲まれた名称は、可変であることを表します。
例:`${dataanyware.home}/server/logs/<日付ディレクトリ>`

3. 全般



3.1. 種別

種別とは、スクリプトの実行時にグローバルリソースの設定を切り替えるための機構です。

従来通りの使用方法に加え、例えば開発/テスト/本番などの各フェーズで、接続先データベースはそれぞれに用意したものを使用したいが、スクリプトは変更したくない、というような場合にグローバルリソースに種別を設定することにより、スクリプトを編集することなくデータベースなどの接続先を切り替えることができます。

また、種別には「デフォルト」という概念があります。これは、ユーザが種別を意識しなくても従来通り使用できるように用意されています。

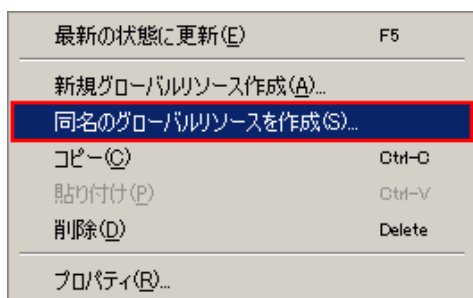
グローバルリソースの新規作成時には「デフォルト」の種別のグローバルリソースのみが作成できます。デザイナーでアダプタのオペレーションアイコンをスクリプトキャンバスに配置し、プロパティで接続先を選択する際に表示されるグローバルリソースは「デフォルト」の種別となります。また、種別を指定しないでスクリプトの実行を行った際にも「デフォルト」のグローバルリソースが使用されます。

初期設定では、これまで通りの使用方法で操作や処理を行うようになっています。設定を変更することにより、種別を指定して同一のスクリプトで複数のグローバルリソースを切り替えて処理を行うことができますようになります。

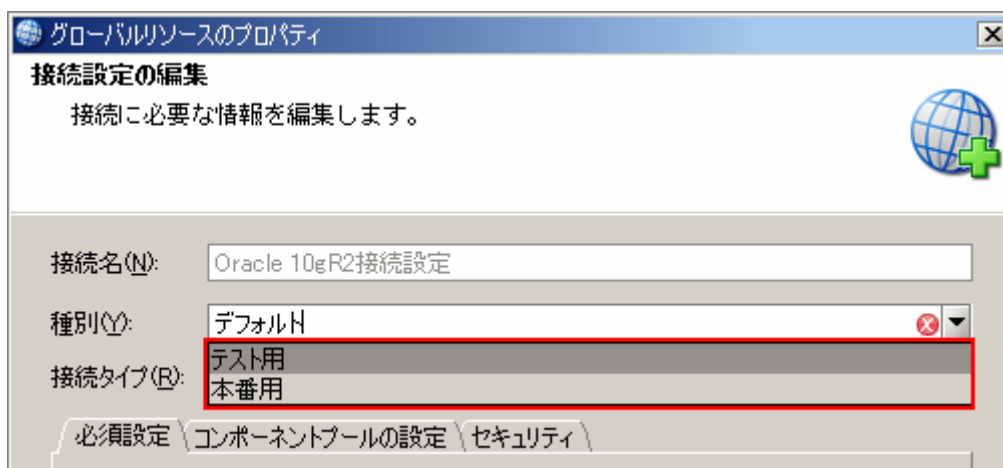
(※)詳細については、Studio のオンラインヘルプを参照してください。

3.1.1. 種別の設定方法

コントロールパネルの「グローバルリソースの設定」で、既存のグローバルリソースの右クリックメニュー [同名のグローバルリソースを作成] から作成します。



グローバルリソースの右クリックメニュー



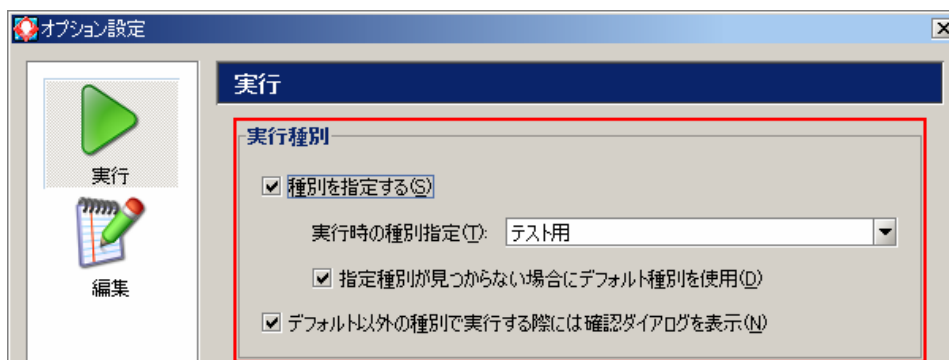
[種別]のプルダウンメニューから作成した種別を選択します



種別はユーザが任意に指定することもできます。

3.1.2. 種別の使用方法

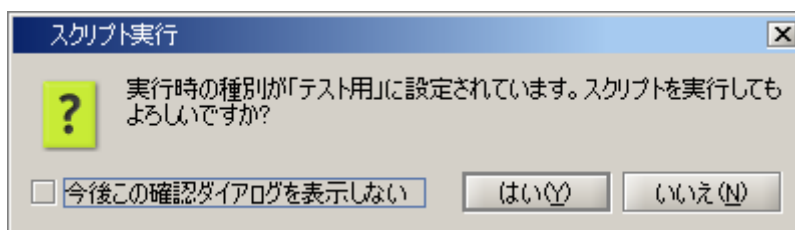
デザイナー



[ツール]-[オプション]の「オプション設定」で設定できます。

(デフォルトは[種別を指定する]にチェックなし=デフォルトの種別で実行になっています。)

実行時に指定した種別のグローバルリソースで処理を行います。



デフォルト以外の種別で実行する際には確認のダイアログが表示されます。

(設定により表示しないようにすることもできます。)

また、指定した種別のグローバルリソースが存在しない場合に、エラーにするかデフォルトの種別のグローバルリソースを使用するかを選択することもできます。

チェックあり:デフォルトの種別のグローバルリソースを使用して実行します。 チェックなし:実行時にエラーが発生します。

トリガー

トリガー設定時の「実行オプションの設定」で指定できます。

(デフォルトは[種別を指定する]にチェックなし=デフォルトの種別で実行になっています。)

ScriptRunner<param>タグ内に以下のように記述します。

```
<param project="root@プロジェクト" script="スクリプト">
  <option key="TYPE">development</option>
  <option key="ENABLE_XML_LOG">true</option>
  <option key="LOG_LEVEL">DEBUG</option>
  <option key="ENABLE_ASSERTION">true</option>
</param>
```

TYPE キーの値と Studio での表示名の関連は、以下の通りです。

表示名	キー値
デフォルト	default
テスト用	test
本番用	production

CLI Console

run コマンドの種別指定オプション「-t」を使用します。

```
# help run
使用方法:

run [ -w | -t <type> | --enable-type-switch ] service script [ args(key=valu e)... ]

# run -t test root@プロジェクトスクリプト

ca0000pg7el1i10o40fid4mshdtq
# run -t default root@プロジェクトスクリプト

ca0000pg7el1i10o40fid4mshdtq
# run -t production root@プロジェクトスクリプト

ca0000pg7el1i10o40fid4mshdtq
```

(※)ScriptRunnerProxy など、その他の詳細については Studio のオンラインヘルプを参照してください。

3.2. HTTPS 通信

Server と各コンポーネントの通信を HTTPS で行うことができるようになりました。

HTTPS 通信が可能なコンポーネントは以下の通りです。

- HTTP トリガー

- Web サービストリガー
- CLI Console
- ScriptRunner
- ScriptRunnerProxy

3.2.1. 設定方法

コントロールパネルの「DataAnyware 2.0Server の設定」から行います。

[セキュリティ]タブの[HTTPS を有効にする]にチェックを入れます。

[ポート番号]および[パスワード]の設定を行い、Server の再起動を行ってください。

ポート番号の初期値は「8443」となります。

キーストアのパスワードを設定します。初期値は「changeit」となります。

キーストアパスワード

HTTPSの暗号化で使用する鍵、証明書が保存されているキーストアのパスワードを設定します。初期値は「changeit」です。キーストアパスワードに対する変更は、DataAnyware 2.0 Server再起動後に有効となります。

パスワード:

4. インストーラ

4.1.Java のバージョンの変更

バンドルしている Java のバージョンを変更しました。

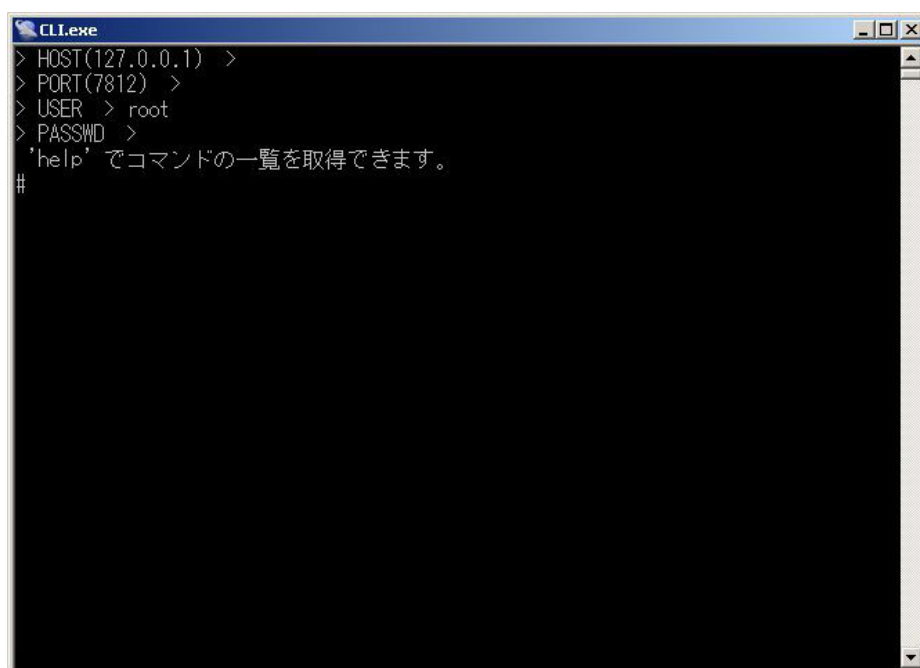
- Sun JRE 1.5.0_12

5. CLI Console

5.1. クライアント CLI



Client に CLI Console を追加しました。



ログインの際にユーザ名とパスワードを入力する必要があります。

(※)ホスト名とポート番号は `boot.properties` に設定されている値がデフォルトで設定されます。

5.2. CLI Console ログ

CLI Console ログ(※)にコマンドを実行したユーザの情報を出力するようにしました。

(※)`${dataanyware.home}/server | client/logs/cli.log`

```
12/20 20:38:37|INFO|ds.console.cli|USER=root|START run -t test root@プロジェクト スクリプト
12/20 20:38:37|INFO|ds.console.cli|USER=root|END run -t test root@プロジェクトスクリプト
12/20 20:38:46|INFO|ds.console.cli|USER=root|START run -t production root@project script
12/20 20:38:46|INFO|ds.console.cli|USER=root|END run -t production root@project script
```

出力例

5.3. 新規コマンド

以下のコマンドを追加しました。

- **proinfo**

プロジェクトの以下の情報を表示するコマンドです。

- プロジェクト名
- 保存場所
- バージョン
- 所有者
- 作成日時
- 更新日時

「-v」オプションを指定することにより、バージョンの一覧を表示することもできます。

- **prodel**

プロジェクトを削除するコマンドです。プロジェクトを削除する場合には、確認を行います。「-f」オプションを指定することにより、確認を行わないで削除することもできます。

- **prodl**

プロジェクトをローカルへダウンロードするコマンドです。デフォルト(オプション指定なし)では、最新バージョンのみダウンロードを行います。また、「-a」オプション指定で「全バージョンの一括ダウンロード」、「-v」オプション指定で「バージョンを指定してダウンロード」を行うことができます。

- **proul**

プロジェクトをローカルからアップロードするコマンドです。デフォルトでは、新規プロジェクトとしてアップロードします。「-e」オプション指定で、上書きコミット(全バージョンの一括ダウンロードしたプロジェクトの場合)または追加コミット(バージョンを指定してダウンロードしたプロジェクトの場合)を行うことができます。

- **deoloy**

プロジェクトをサービスとしてサーバに登録します。

- **undeploy**

サーバに登録されたサービスを削除します。

- **myservices**

サービスの一覧を表示します。(projectnames と同一機能です。)

- **myprojects**

現在ログイン中のユーザが参照できるプロジェクト一覧を表示します。

5.4. 追加オプション

以下のコマンドにオプションを追加しました。

- mount

マウント先の設定を変更することができる「-m」オプションを追加しました。

コマンド例:

```
# mount -m /<old mount> /<new mount>
```

- run

種別の指定を行う「-t」オプションを追加しました。

コマンド例:

```
# run -t production root@プロジェクト スクリプト
```

- grp

種別の指定を行う「-t」オプションを追加しました。

コマンド例:

```
# grp -c Oracle10gR2 接続設定 -t production
```

6. DataAnyware 2.0 Server

6.1. RMI 通信ポートの固定化機能

サーバとクライアントの受信の RMI 通信ポートを固定できる機能を追加しました。これにより、Server と Studio 間のファイアウォールを越えることができるようになります。

まず。

【設定方法】

`${dataanyware.home}/server | client/conf/system.properties` ファイルの「ds.rmi.port」プロパティにポート番号を記述して、Server と Studio を再起動してください。

6.2.XML ログへのバージョン情報の出力

XML ログにバージョン情報を出力するようにしました。

XML ログ(`${dataanyware.home}/server/logs/<日付ディレクトリ>/execution/<実行 ID>.xml`)内の

```
<executionStart time="1198148082834" production-version="2.3.0" />
<scriptStart time="1198148082943" id="ak00005g76l1i0uo6fevqf3qhdt" name="スクリプト"
type="default" enableTypeSwitch="true" repository="/home/root/projects/プロジェクト"
supportsLargeData="false" psp="false" />
```

「executionStart」要素の「production-version」属性に出力されます。

また、指定した種別も「scriptStart time」要素の「type」属性に出力されます。

(※)詳細については Studio のオンラインヘルプを参照してください。

7. ScriptRunner

7.1. 起動設定ファイルのパスワード暗号化

起動設定ファイル内のパスワードの暗号化機能を追加しました。

`${dataanyware.home}/<server | client>/bin` フォルダ下にある `ScriptRunner.exe` を、`-e` オプションを指定して実行します。

```
>ScriptRunner.exe -e <起動設定ファイル> <暗号化後のファイル名>
```

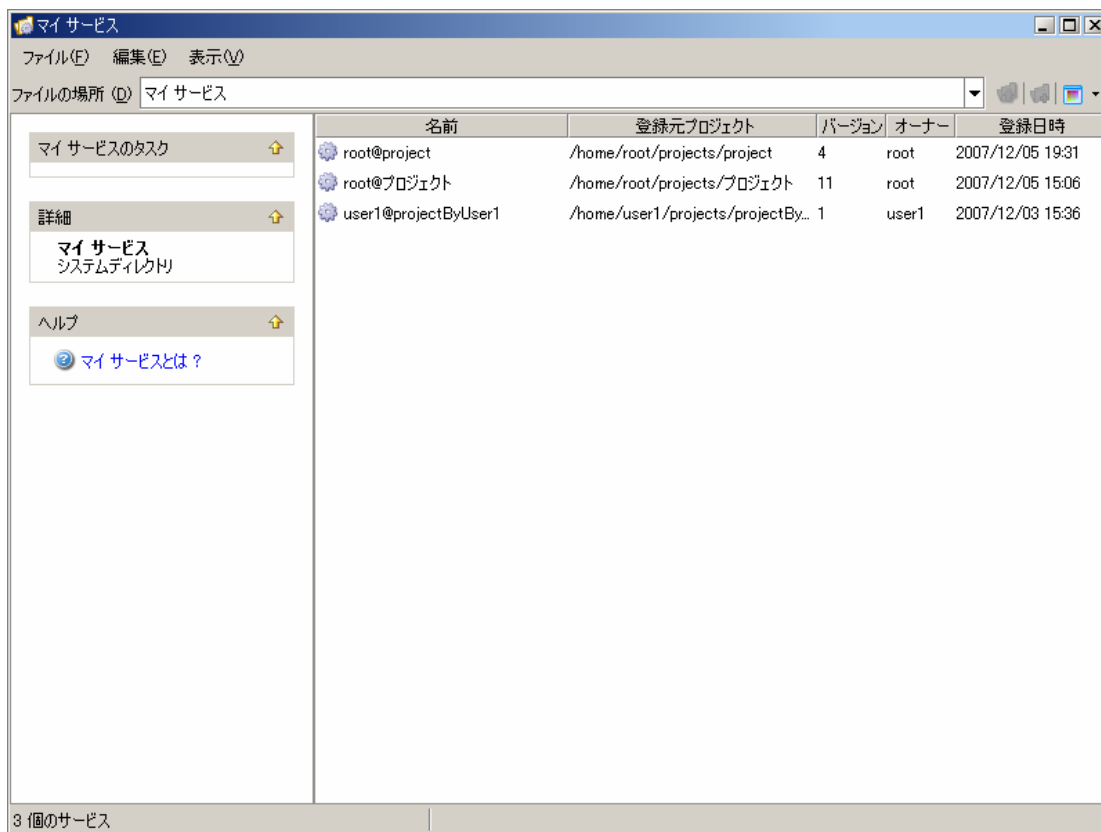
8. Studio

8.1. マイサービス

サービス(サーバに登録されたプロジェクト)の管理を行うツール「マイサービス」を追加しました。



Studio のデスクトップのアイコン また、これまではサービスを削除する場合にはデザイナのメニューから行っていましたが、1.2.0 からはマイサービスで行うことになります。



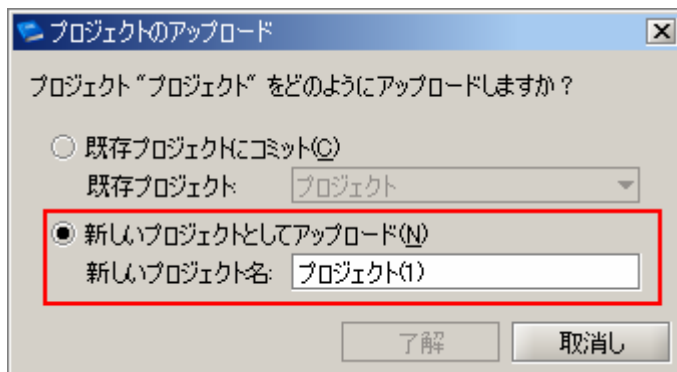
「マイサービス」画面

(※)詳細については Studio のオンラインヘルプを参照してください。

8.2. マイプロジェクト

8.2.1. プロジェクトアップロード時のデフォルト名付与

プロジェクトのアップロード時に、「プロジェクトのアップロード」ダイアログの[新しいプロジェクト名]にプロジェクト名を表示するようにしました。



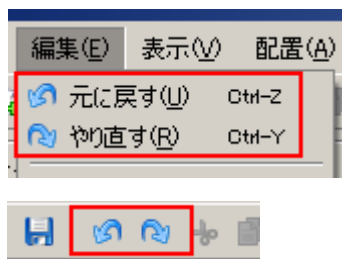
ダウンロード時のプロジェクト名が表示されます。同名のプロジェクトが存在する場合は、数字を付与します。例:「プロジェクト(1)」

8.3. デザイナ

8.3.1. Undo/Redo

デザイナ上の操作を Undo/Redo できるようになりました。

メニューバーの[編集]-[元に戻す]/[やり直す]、ショートカットキーの Ctrl+Z、Ctrl+Y、またはツールバーのアイコンで Undo/Redo を行うことができます。



操作履歴はスクリプト単位で保持されます。(エディタ単位では保持されません。)また、操作履歴はプロジェクトを保存するまで保持し続けます。

- Undo/Redo の対象と操作

- スクリプト変数の新規作成/変更/移動/削除
- スクリプトキャンバスでの操作(アイコンの配置/移動/削除/設定変更/フローの設定など)
- Mapper エディタでの操作(アイコンの配置/移動/削除/スキーマ編集/設定変更/マッピングの設定など)
- HTML ビルダーでの編集操作(テキストの入力/スキーマ編集など)
- プロパティンスpekタでの設定変更(スクリプト変数/コンポーネント/Mapper ロジックなど)
- エディタの開閉

プロパティウィンドウ上では Undo/Redo を行うことはできません。[完了]ボタン押下で設定が反映され、Undo/Redo の対象となります。

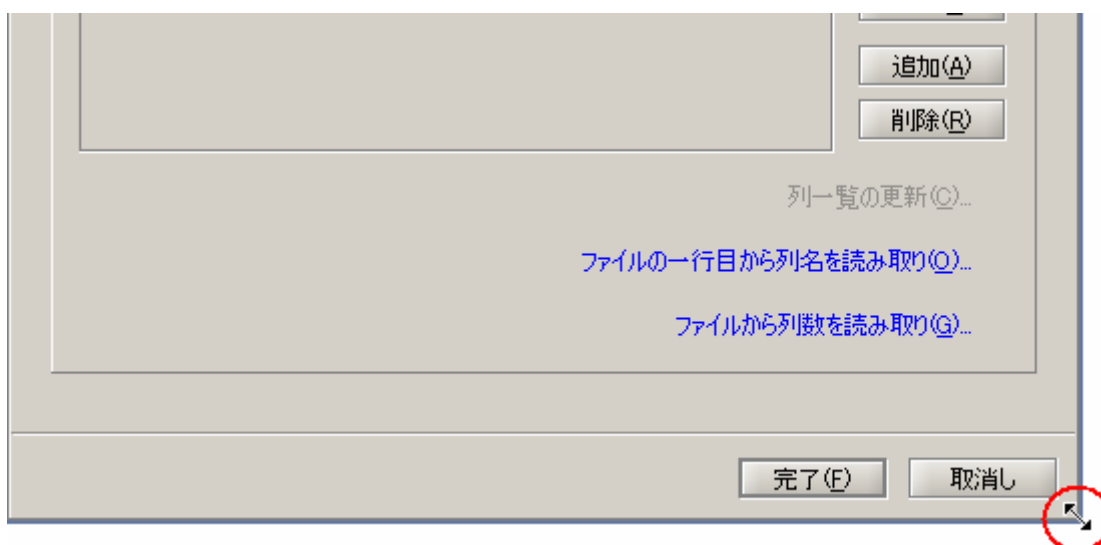
- 以下の操作は Undo/Redo の対象となりません。

- プロジェクトの新規作成/削除
- プロジェクト名の変更
- スクリプトの新規作成/削除/移動
- スクリプトの設定(スクリプト名、データ処理方式など)
- フォルダの新規作成/削除/移動
- リアルタイムプレビュー機能の設定

- ユーザ定義ロジックの新規作成/削除
- ユーザ定義ロジックの編集
- グローバルリソースの設定
(※アダプタでグローバルリソースを選択する操作は Undo/Redo 対象です)
- スクリプトキャンバスのグリッド表示
(※Mapper キャンバスのグリッド表示は Undo/Redo 対象です)
- 画面構成の変更

8.3.2. プロパティウィンドウのサイズ変更

コンポーネントのプロパティウィンドウのサイズを大きくできるようにしました。



プロパティウィンドウの右下にマウスカーソルを合わせると、ウィンドウサイズを大きくすることができます。

8.3.3. プロパティウィンドウのサイズ自動拡張機能

- コンポーネントのプロパティ内のスキーマ定義が 6 行以上の場合、プロパティウィンドウのサイズを自動的に拡張し、一度に表示させる行を増やすことができますようにしました。

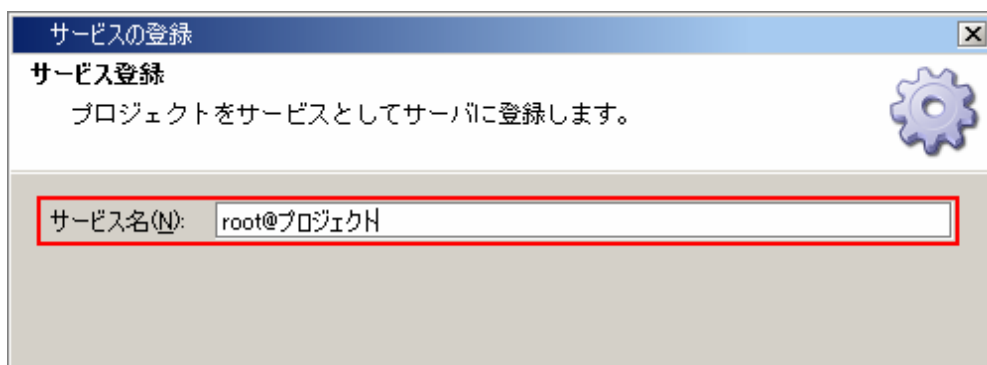
最大で 15 行まで表示させることができます。

- 複数行入力が可能なフィールドで、6 行以上の文字を入力した場合、プロパティウィンドウのサイズを自動的に拡張し、一度に表示させる行を増やすことができますようにしました。

最大で 30 行まで表示させることができます。

8.3.4. サービス名の指定

プロジェクトをサービスとして登録する際に、サービス名を指定できるようになりました。



サービス名をユーザが任意に指定できます。また、登録したサービスは「マイサービス」で管理を行います。

8.3.5. プロジェクト/スクリプト仕様書の生成

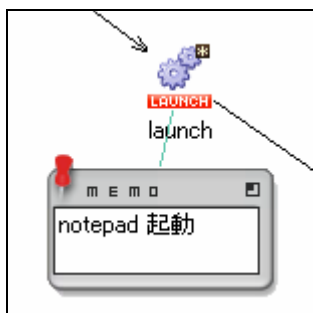
プロジェクト/スクリプトの仕様書生成で、出力された仕様書にスクリプトの設定(トランザクション属性/大容量データ処理)およびコンポーネントに関連付けられたメモの情報を出力するようにしました。

1. スクリプト

- 更新日時
2007年12月14日
- スクリプト変数
入力

No.	変数名	型	入力	出力	初期値
- トランザクション属性
REQUIRED
- 大容量データ処理
大容量処理を行う

スクリプトの設定(トランザクション属性/大容量データ処理)を出力します。



コンポーネントに関連付けられた(フローが引かれた)メモを作成します。

・コンポーネント一覧

No.	名前	種別	入力データ	関連メモ
1		start	なし	なし
2	launch	外部アプリケーション起動	なし	notepad 起動
3		end	なし	なし
4		メモ	なし	なし

コンポーネントに関連付けられた(フローが引かれた)メモの内容をコンポーネント一覧の「関連メモ」の項に出力します。

8.3.6. スレッド処理



マルチスレッドで処理を行うことができる「スレッド処理」を追加しました。

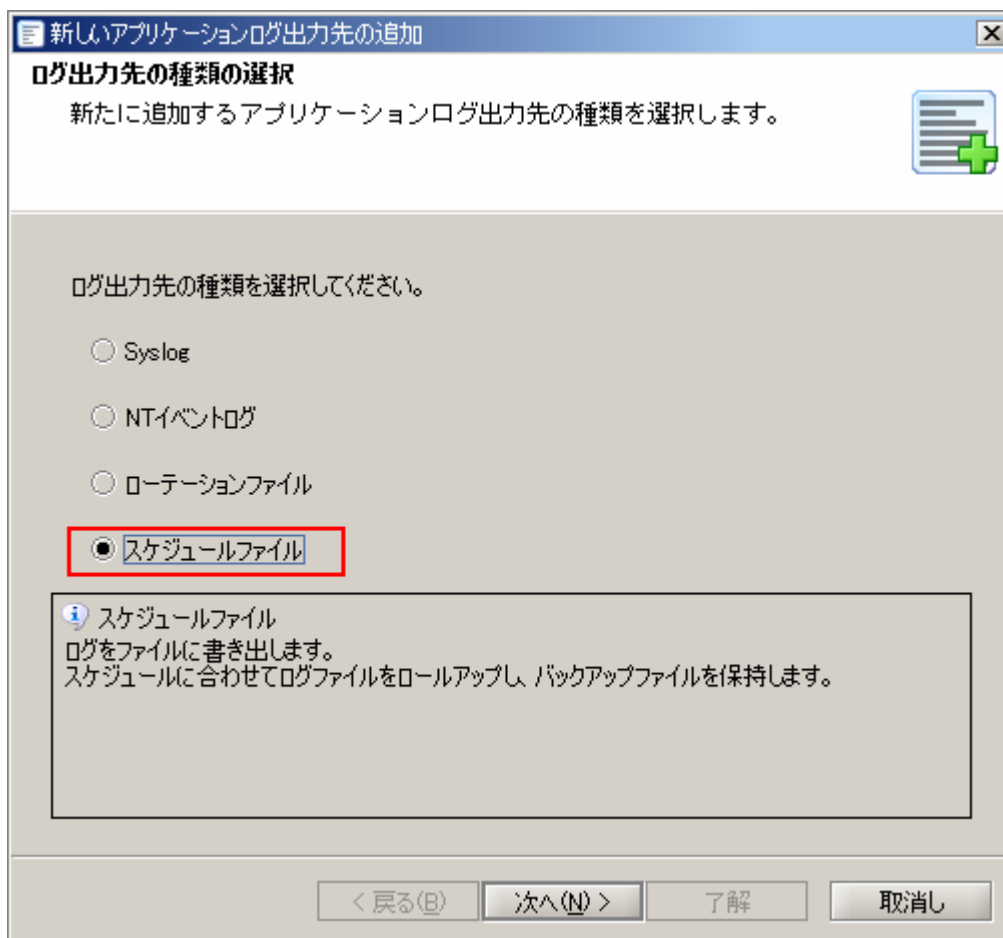
8.3.7. 待機処理のミリ秒指定

待機処理で、[待機時間]をミリ秒まで指定できるようにしました。

9. コントロールパネル

9.1. アプリケーションログ出力先設定

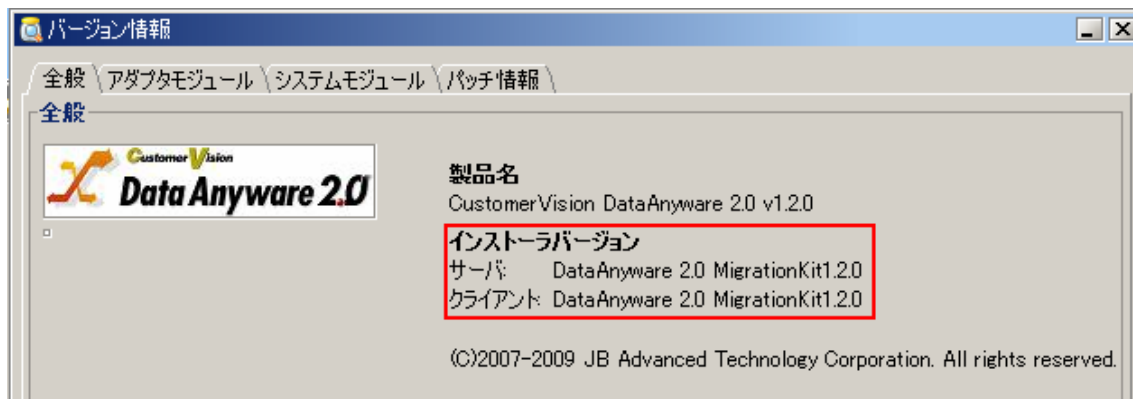
日/週/月単位でログファイルのロールアップを行う「スケジュールファイル」機能を追加しました。



スケジュールに合わせてログファイルをロールアップし、バックアップファイルを保持します。

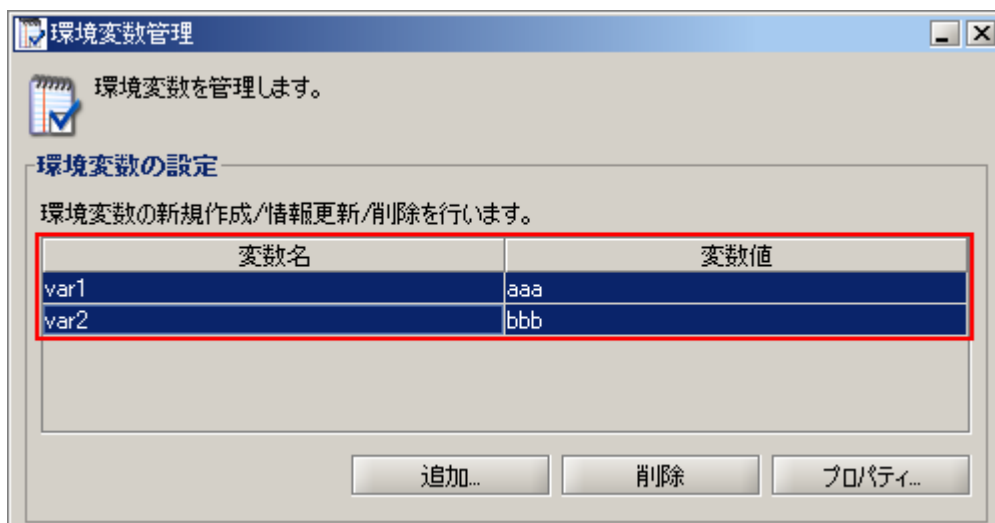
9.2. バージョン情報

[全般]タブに、Server および Client の「インストーラバージョン」を表示するようにしました。



9.3. 環境変数管理

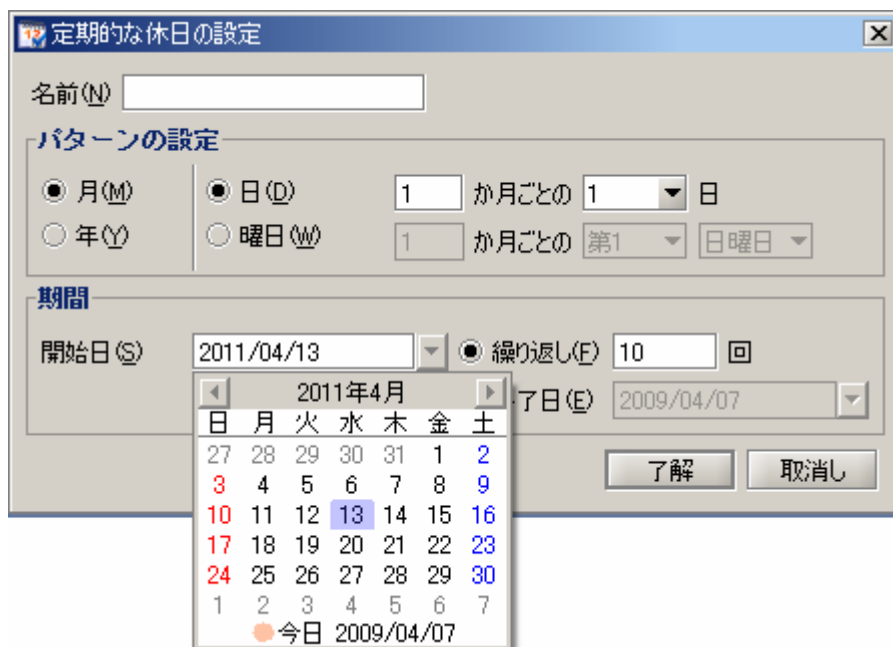
環境変数の複数選択削除をできるようにしました。



Ctrl キーを押下しながら選択で一つずつ、Ctrl+A キーで全選択となります。

9.4. 休日設定

「基準カレンダー」に 2011 年から 2020 年までのカレンダーを追加しました。

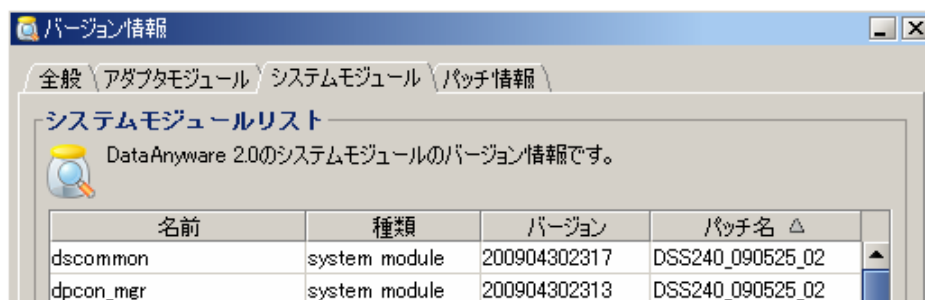


9.5. 「DataAnyware 2.0 Server の設定」パッチ情報タブ

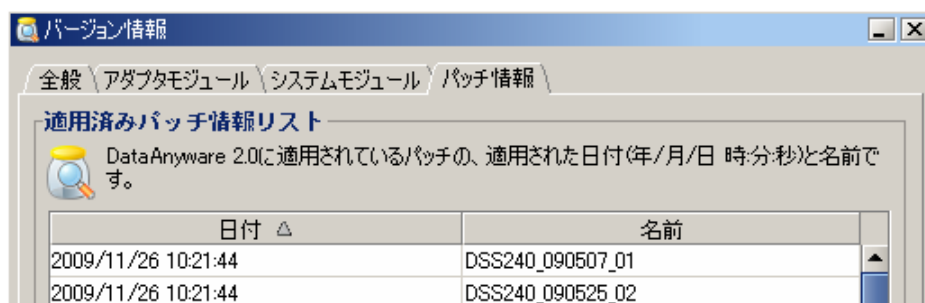
「バージョン情報」のモジュールリストに、適用済みパッチ番号が表示される[パッチ番号]項目と、適用済みパッチの一覧が表示される[パッチ情報]タブを追加しました。



[アダプタモジュール]タブ



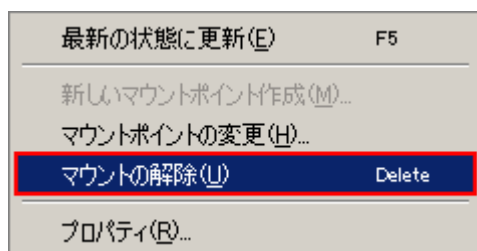
[システムモジュール]タブ



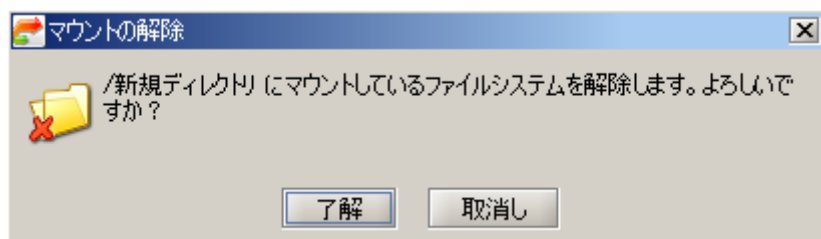
[パッチ情報タブ]では適用したパッチの一覧が表示されます。

9.6. 「マウント設定」 マウント解除時のオプション

マウントの種類が「拡張ローカルファイルシステム」または「データベースファイルシステム」の場合、[マウントの解除]時にリポジトリ DB 内の関連するデータを削除する機能を追加しました。



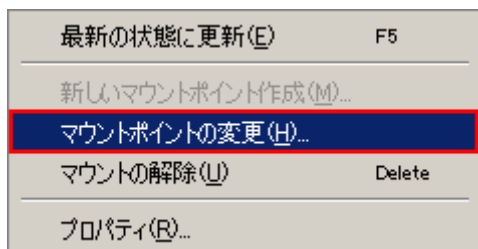
マウントを選択して右クリックメニューを開きます。



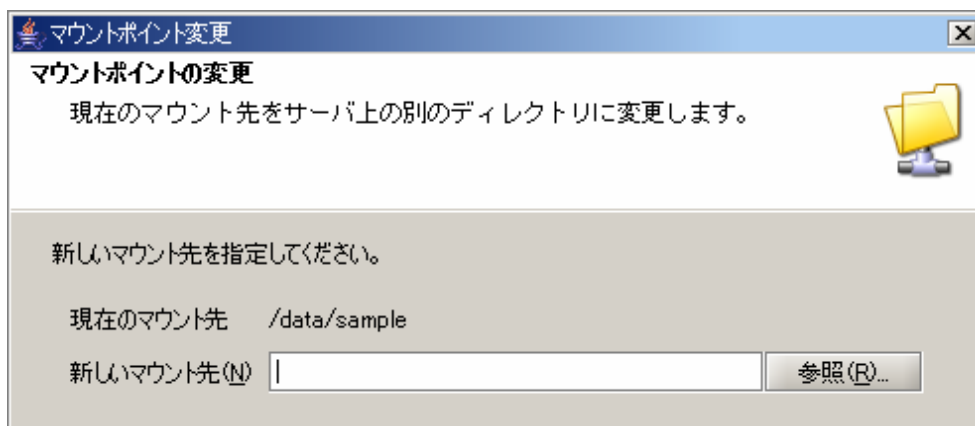
確認のダイアログが表示されます。[了解]ボタンを押下するとリポジトリ DB 内の関連するデータが削除されます。

9.7. 「マウント設定」 マウントの変更

マウント先の設定を変更することができる[マウントポイントの変更]機能を追加しました。



マウントを選択して右クリックメニューを開きます。



新しいマウント先を入力または選択します。

9.8. 「グローバルリソースの設定」 グローバルリソースの ACL 機能

グローバルリソースに ACL 機能を追加しました。

グローバルリソースのプロパティ

接続設定の編集
接続に必要な情報を編集します。

接続名(N): Oracle 10gR2接続設定

種別(O): デフォルト

接続タイプ(R): Oracle (Oracle 10gR2接続設定)

必須設定 コンポーネントプールの設定 **セキュリティ**

所有者: root

グループ: root

アクセス権限:

	リソースの使用	リソースの編集
所有者	☒	☒
グループ	☒	☐
その他	☒	☐

[セキュリティ]タブから設定できます。

10. マイログ

10.1. プロジェクト名やスクリプト名での検索機能

ログ検索条件にプロジェクト名やスクリプト名を指定して検索を行うことができる機能を追加しました。

※トリガーや ScriptRunner などの外部ツールから実行したスクリプトが対象となります。

(デザイナーから実行したスクリプトは検索対象になりません。)

☐ サービス名やスクリプト名を限定する(P)

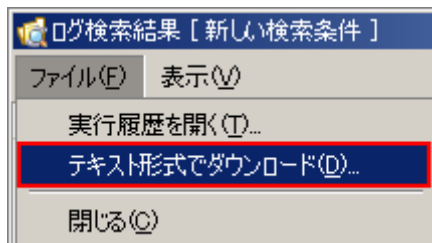
サービス名:

スクリプト名:

ログ検索条件の作成/変更時に指定が可能です。

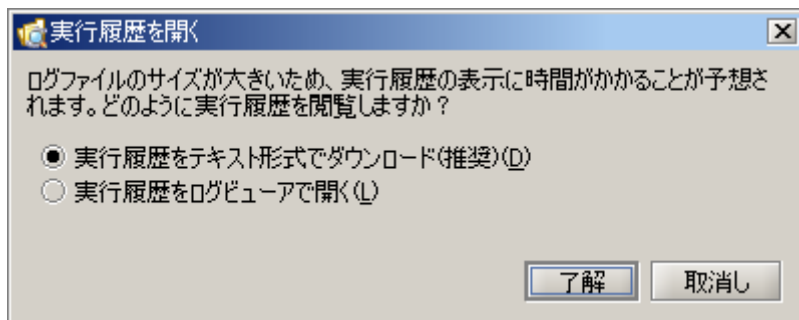
10.2. ログのダウンロード機能

ログ(XML ログ)をテキスト形式でダウンロードすることができる[テキスト形式でダウンロード]メニューを追加しました。



対象のログを選択して[ファイル]-[テキスト形式のダウンロード]を選択します。

XML ログのサイズが閾値以上の場合は、検索結果をダブルクリックした際にテキスト形式でダウンロードするか従来通りログビューアを開くかを選択できるダイアログが表示されます。



選択ダイアログが表示される閾値は以下のプロパティで定義されています。

【配置場所】

`${dataanyware.home}/client/conf/system.properties`

【キー】

`mylogs.download.threshold.size`

【デフォルト値】

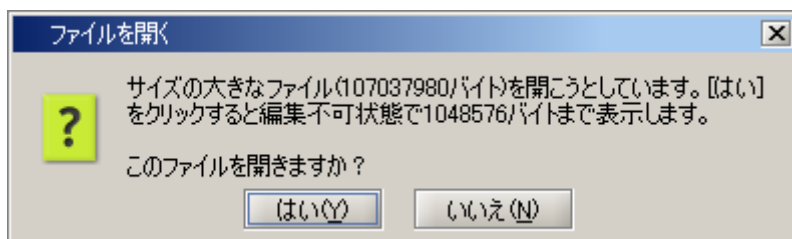
131072 (128KB)

※プロパティが設定されていない場合のデフォルト値も「131072」(128KB) です。

11. エクスプローラー

11.1. テキストエディタで開くことのできるファイルサイズの指定

テキストエディタで開くことのできるファイルサイズをプロパティで指定できるようにしました。設定値より大きいサイズのファイルを開くと、確認のダイアログが表示されます。



[[はい]]を押下した場合は、

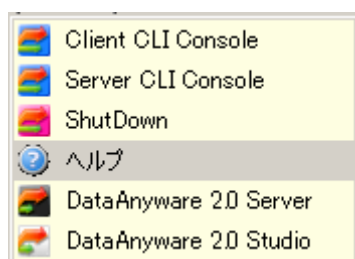
- ・ ファイルの先頭から設定値分の内容を表示
- ・ 開かれたテキストエディタは編集(保存)不可

となります。

12. ヘルプ

12.1. ブラウザヘルプの対応

ウェブブラウザから参照可能な「ブラウザヘルプ」を追加しました。



Windows スタートメニューから、「DataAnyware 2.0」-「ヘルプ」を選択します。

デフォルトのウェブブラウザが起動し、ヘルプが表示されます。

 A screenshot of the DataAnyware 2.0 Help page displayed in Internet Explorer. The browser window title is 'CustomerVision DataAnyware 2.0 ヘルプ - Windows Internet Explorer'. The address bar shows 'C:\Program Files\DataAnyware\client\doc\help\index.html'. The page content includes a table of contents on the left, a main heading 'DataAnyware 2.0ヘルプ', and a section titled 'ヘルプの起動方法' (How to start the help). Below this is a table with columns 'ツール' (Tool), '方法' (Method), and '補足' (Notes). The table lists methods to start the help for 'DataAnyware 2.0 Studio' and 'ブラウザ' (Browser). There is also a section titled '検索について' (About search) and a search bar.

ツール	方法	補足
DataAnyware 2.0 Studio	スタートメニューから【ヘルプ】を選択する	
	デスクトップにあるヘルプアイコン  をクリックする	
	デザイナのメニューバーから【ヘルプ】を選択する	
	デザイナ上でF1キーを押下する	
ブラウザ	Windowsスタートメニューの「すべてのプログラム」-「DataAnyware 2.0 <バージョン> <パッケージ>」-「ヘルプ」を選択する	推奨ブラウザ: Internet Explorer Version 6.0以上

12.2. オンラインヘルプ「同期ボタンの追加」

デザイナ上で選択したオペレーションのヘルプを同期して表示するかどうかを選択できる「同期ボタン」を追加しました。



デフォルトではオン状態となり、デザイナ上で選択したオペレーションと同期してヘルプを表示します。



オフ状態ではデザイナと同期せず、ヘルプで開いたページを表示します。

12.3. 新規追加の項目

以下の項目を新規追加しました。

- DataAnyware 2.0 ログガイド

本製品で出力されるログについてのガイドです。

- 種別

1.2.0 から導入された種別の概念に関する項目です。

- データ型

内部で使用されているデータ型についてのガイドです

- トラブルシューティング

本製品を使用するにあたってのトラブルシューティングガイドです。

- グローバルリソース

グローバルリソースの概念と動作に関する項目です。

- トランザクション

トランザクションの概念と機能に関する項目です。 その他オンラインヘルプ全体のページに対し、記述フォーマットの統一、文言の統一、不足内容の追加などを行っています。

13. DB トリガー



13.1. DB トリガーとは

DB トリガーとは、指定したデータベースのテーブル内のステータスカラムを監視し、状態変化時に発火します。例えば、複数のデータベース/アプリケーションの同期処理を行う場合などに使用します。

DBトリガー

DBトリガーの設定

指定したカラムのステータスによりスクリプトを起動します。

トリガー名(T):

新規DBトリガー

接続(R):

Oracle 10gR2接続設定

監視テーブル(W):

DBTRIGGER_TEST

ステータスカラム(C):

COL3

監視イベント(E):

☒ 挿入時 ☐ 更新時 ☐ 削除時

監視間隔(秒)(I):

10

終了時オプション

☒ 成功時のみステータスを変更する(S)

☐ 成功/失敗にかかわらずステータスを変更する(D)

ステータス値:

値	状態
0	未変更状態
1	挿入時トリガー待ち状態
2	更新時トリガー待ち状態
3	削除時トリガー待ち状態
4	削除状態

< 戻る(B)

次へ(N) >

完了(F)

取消し

13.2. DB トリガーの使用方法

- ステータスカラムの作成

まず、監視するテーブルにステータスカラムを追加します。DB トリガーはこのカラムの値を監視します。デフォルトでは、値は以下のように定義されています。(任意に変更可能)

値	状態	説明
0	未変更状態	挿入および更新イベント発火後の状態です。
1	挿入時トリガー待ち状態	監視イベントが挿入時の場合に、トリガーが発火する状態です。
2	更新時トリガー待ち状態	監視イベントが更新時の場合に、トリガーが発火する状態です。
3	削除時トリガー待ち状態	監視イベントが削除時の場合に、トリガーが発火する状態です。
4	削除状態	削除イベント発火後の状態です。

- スクリプトの作成
- DB トリガーの作成

13.3. DB トリガーの仕様

DB トリガーの動作は下記のようにになっています。

- トリガーの発火時に、監視イベントのステータス値を持つステータスカラムに対し、システム側で一意的な値に更新します。トリガーの終了時には、一意的な値を持つステータスカラムに対してのみ[未変更状態]または[削除状態]に変更します。
- 監視対象のステータスカラムのデータ型は整数型(INTEGER)である必要があります。
- [ステータス値]で指定できる値はデフォルトでは「0」以上「1000」未満および「11000」以上の整数です。(指定できる最大値はステータスカラムのデータ型に依存します。)
「1000」以上、「11000」未満の整数は「実行 ID」として内部でトリガーごとに割り当てられます。

※詳細については、DB トリガーのヘルプ「実行 ID について」を参照してください。

14. Mapper 全般

14.1. 内部演算の仕様変更

Mapper の内部演算を全て BigDecimal 型で行うようにしました。

本対応により、以下のようなケースの動作が変更となります。

【小数点以下が切り捨てられるケース】

例: 文字列「1.1」 + 整数型「1」

対応前: 「2」

対応後: 「2.1」

※対応前では、文字列の「1.1」が内部で int 型として扱われ、小数点以下が切り捨てられる。

【桁あふれが発生するケース】

例: 文字列型「1」 + 整数型「2147483647」

対応前: 「-2147483648」

対応後: 「2147483648」

※対応前では、文字列型と整数型の演算のため内部で int 型として扱われる。演算結果が int の範囲を超えたため、範囲内に数値が丸められる。

【丸めの誤差が発生するケース】

例: 10 進数型の「0.1」 + Double 型の「0.1」

対応前: 「0.20000000000000000555111…」

対応後: 「0.2」

※Double 型の「0.1」の処理方法に問題があり、浮動小数点数誤差が含まれる形で演算を行っていた。

【浮動小数点数誤差の影響により期待した計算結果にならないケース】

例: 10 進数型の「1」 ÷ Double 型の「0.1」

対応前: 「9」

対応後: 「10」

※Double 型の「0.1」の処理方法に問題があり、浮動小数点数誤差が含まれる形で演算を行っていた。

【小数点数精度が統一されていないケース】

例 1: Double 型の「0.5」 + Double 型の「0.5」

対応前: 「1」

対応後: 「1.0」

例 2: 10 進数型の「0.5」 + 10 進数型の「0.5」

対応前: 「1.0」

対応後: 「1.0」

※対応前は double 型の演算を BigDecimal 型で行っていなかったため、scale の情報を消失していた。

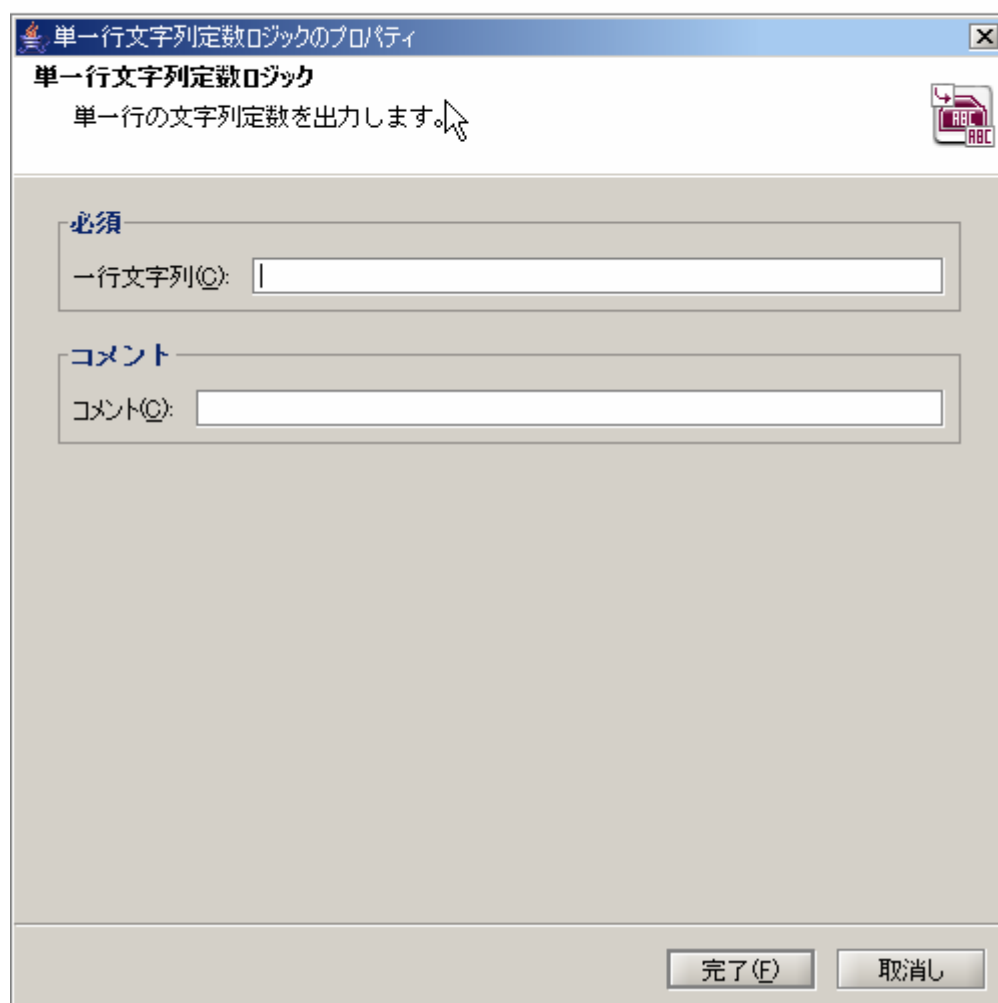
14.2. 「条件」 - 「数値」カテゴリの **Mapper** ロジックの仕様変更

[条件] - [数値]カテゴリの Mapper ロジックの演算を、数値として比較するようにしました。例えば、「1.0」と「1」は等しいとみなされます。

14.3. Mapper ロジックの編集ダイアログ

Mapper ロジックのプロパティの編集を、マッピングキャンバス上から行うことができるようにしました。

Mapper ロジックをマッピングキャンバスに配置し、アイコンのダブルクリックで編集ダイアログを開くことができます。



単一行文字列定数ロジックのプロパティ

単一行文字列定数ロジック

単一行の文字列定数を出力します。

必須

一行文字列(C):

コメント

コメント(C):

完了(F) 取消し

14.4. プロパティの追加

「数値」 - 「変換」カテゴリの null/空文字列を変換ロジックに、以下のプロパティを追加しました。

- [null の変換] (デフォルトはチェックあり)
- [空文字の変換] (デフォルトはチェックあり)



null/空文字の変換ロジックのプロパティ

null/空文字の変換ロジック
入力値がnull/空文字の場合に、規定値(数値)に置き換えます

必須

nullの変換: ☒

空文字の変換(B): ☒

規定値(B):

コメント

コメント(C):

14.5. null オプション

以下の Mapper ロジックに、null を入力値とした場合の動作を指定できる「null オプション」プロパティを追加しました。

「条件」-「真偽」

And 演算、Or 演算、Not 演算、真偽値に変換

「条件」-「数値」

等しい、等しくない、未満、以下、より大きい、より小さい、以上

「条件」-「数値」

同じ、同じでない、始まりが同じ、終わりが同じ、含む、正規表現にマッチ

「条件」-「日付」

同一日時、以前、以後、日時が存在

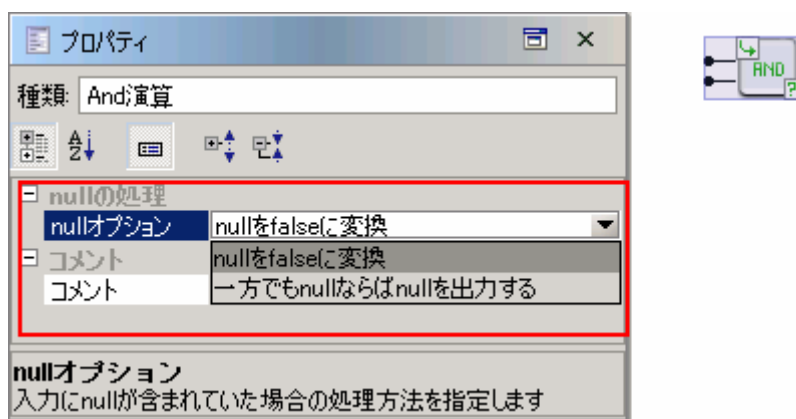
「数値」-「演算」

足し算、引き算、掛け算、割り算、余り

「文字列」-「演算」

連結

14.5.1. オプションプロパティの使用方法



Mapper アイコンにフォーカスをあわせてプロパティインスペクタからオプションを選択します。

注意

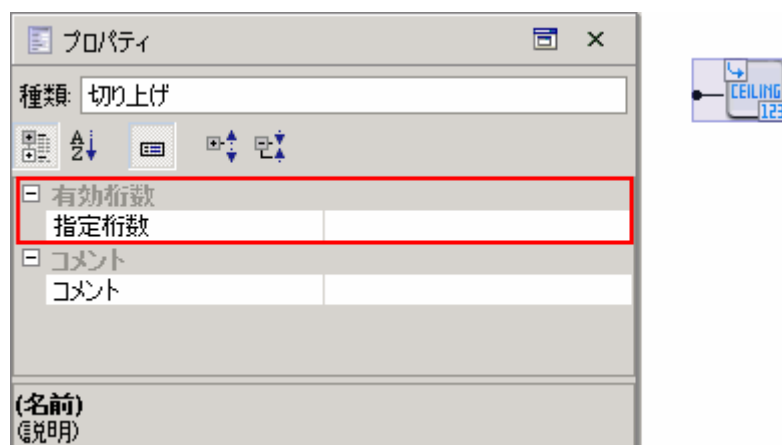
1.1.0 からアップグレードした場合のデフォルトは、[一方でも null ならば null を出力する]となります。
各プロパティの動作の詳細については、オンラインヘルプを参照してください。

14.6. 指定桁数オプション

以下の Mapper ロジックに、小数点以下の桁数を指定できる[指定桁数]オプションプロパティを追加しました。

「数値」-「関数」 切り上げ、切り捨て、四捨五入

14.6.1. オプションプロパティの使用方法



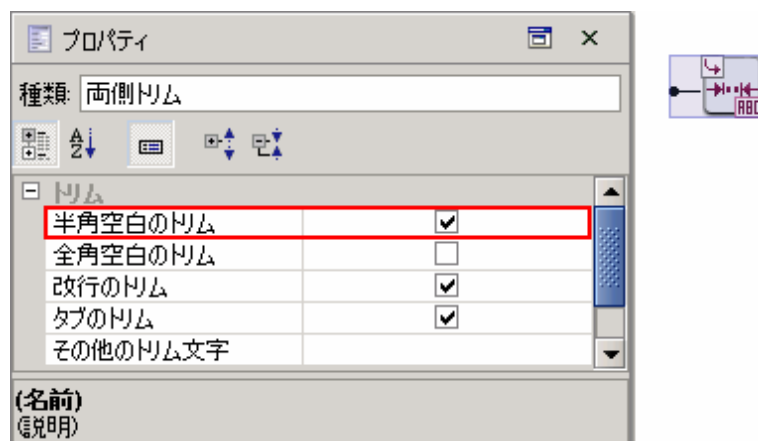
Mapper アイコンにフォーカスをあわせてプロパティインスペクタからオプションを選択します。各プロパティの動作の詳細については、オンラインヘルプを参照してください。

14.7. トリムオプション

以下の Mapper ロジックに、半角空白をトリムするかどうかを選択できる[半角空白のトリム]オプションプロパティを追加しました。デフォルトはチェックあり(トリムする)となります。

「文字列」-「トリム」 両側トリム、左側トリム、右側トリム

14.7.1. オプションプロパティの使用方法



Mapper アイコンにフォーカスをあわせてプロパティインスペクタからオプションを選択します。各プロパティの動作の詳細については、オンラインヘルプを参照してください。

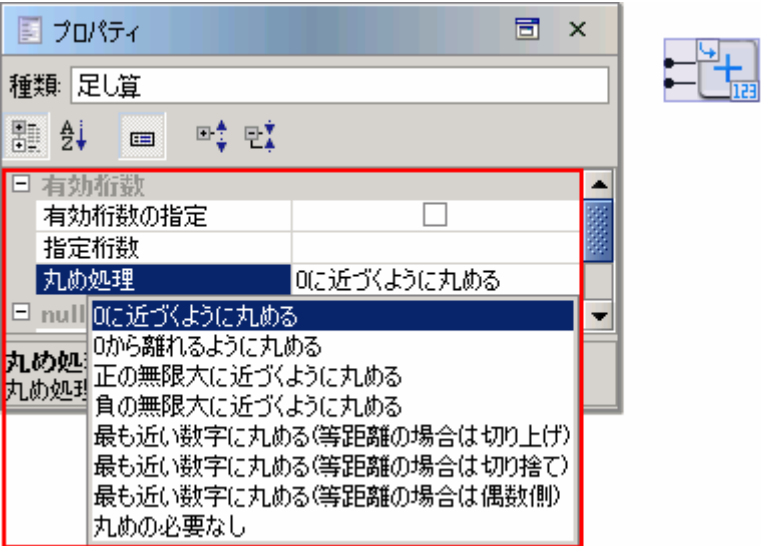
14.8. 有効桁数の指定と丸め処理

以下の Mapper ロジックに、有効桁数の指定と丸め処理の有無を選択できるオプションプロパティを追加

しました。デフォルトは[有効桁数の指定]はチェックなし、[丸め処理]は[0 に近づくように丸める]となります。

「数値」 - 「演算」 足し算、引き算、掛け算、余り

14.8.1. オプションプロパティの使用方法



Mapper アイコンにフォーカスをあわせてプロパティインスペクタからオプションを選択します。

各プロパティの動作は以下の通りです。

プロパティのキー	説明
0 に近づくように丸める	絶対値を切り捨てます。
0 から離れるように丸める	絶対値を切り上げます。
正の無限大に近づくように丸める	正の数値は切り上げます。負の数値は絶対値を切り上げ、符号は変わりません。
負の無限大に近づくように丸める	正の数値は切り捨てます。負の数値は絶対値を切り上げ、符号は変わりません。
最も近い数字に丸める (切り上げ)	4 捨 5 入します。
最も近い数字に丸める (切り捨て)	5 捨 6 入します。
最も近い数字に丸める (偶数)	指定桁数の位置にある数値が偶数なら切り捨て、奇数なら切り上げます
丸めの必要なし	丸めが必要な場合は、実行時にエラーが発生します。

15. Mapper

15.1. 新規追加された Mapper ロジック



「数値」 - 「集合」に以下の Mapper ロジックを追加しました。



合計値 入力ノードの合計値を出力します。



平均値 入力ノードの平均値を出力します。



ノードの数 入力ノードの数を出力します。



ノードの最大値 入力ノードの最大値を出力します。



ノードの最小値 入力ノードの最小値を出力します。

詳細については、オンラインヘルプを参照してください。

16. アダプタ全般

16.1. パスワードの暗号化

アダプタやグローバルリソースで設定されるパスワード情報の暗号化保存に対応しました。

17. データベースアダプタ

17.1. テーブル書き込み処理および更新系 SQL 実行処理のエラー時の詳細なダンプファイルの出力機能

テーブル書き込み処理および更新系 SQL 実行処理で、バッチ更新(※1)処理実行でのエラー発生時に詳細なダンプファイルを出力するようにしました。

(※1)バッチ更新とは

バッチ更新とは複数の SQL 文を一度にデータベースに送り、処理させる機能です。一つ一つ SQL 文を実行

する場合に比べ、パフォーマンスが向上します。

テーブル書き込み処理では、[キーの値と一致する行は更新する]にチェックを入れていない状態で書き込み処理を行った場合に、バッチ更新でデータを書き込みます。

更新系 SQL 実行処理では、[SQL 文]で「SQL パラメータ」を使用した場合に、バッチ更新でデータを書き込みます。

データベースへバッチ更新処理でデータ書き込み時にエラーが発生した場合、

`${dataanyware.home}/server/logs/<日付ディレクトリ>/execution/<実行 ID ディレクトリ>/dump.xml`

ファイルの「failureResults」要素内にエラーの詳細内容が出力されます。(※2)

(※2)書き込み先のデータベースやデータ型によっては、バッチ更新処理が行われません。出力される内容は以下の通りです。

result 要素の key 属性値	説明
dump_error_message	エラー発生時のダンプ情報を出力します。 [SQL] エラー発生時に実行した SQL 文です。 [Error Records Range] 入力データでエラーになった箇所の開始行番号と 終了行番号です。 [Batch Insert Count] バッチ更新処理に成功したレコード数です。 [Batch Size] バッチサイズです。 [Error Records] エラー発生時にバッチ更新処理で用いられたデータを出 力します。入力データの Error Records Range 分が出力されます
error_code	SQL 例外(SQLException)が発生した場合のエラーコードです。

18. DB2 アダプタ

18.1. DB2 V9.5 アダプタの追加

DB2 V9.5 アダプタは、DB2 V9.5 に接続可能なアダプタです。

19. Oracle アダプタ

19.1. Oracle 11g アダプタの追加

Oracle 11g アダプタは、Oracle 11g R1 に接続可能なアダプタです。

20. SQL Server アダプタ

20.1. ロードによる書き込み処理の追加

SQL Server 2005 JDBC アダプタにロードによる書き込み処理を追加しました。
ロードによる書き込み処理はデータベースのクライアントの SQL ロードを使用するため、通常の書き込み処理と比べ高速に処理を行うことができます。

ロードによる書き込み処理のプロパティ

ロードによる書き込み処理
ロードによる書き込み処理のプロパティを設定します。

名前(N): put_table_by_loader

必須設定 コメント

接続先(O): SQL Server 2005接続設定

テーブル名(T): TEST

☐ テーブル名を引用符で囲う(E)

データファイル(R): C:\data\datafile

フォーマットファイル(M): C:\format\formatfile

区切り文字(D): .

スキーマ定義

列名	型
KEY1	varchar (String)
KEY2	varchar (String)
VALUE	varchar (String)

テーブル情報の表示(Y)...

完了(F) 取消し

※Server が稼動している OS に SQL Server 2005 クライアント(管理ツール)がインストールされている必要があります。

20.2. Windows 統合認証の対応

SQL Server 2005 JDBC アダプタに、Windows 統合認証を使用して接続を行うことができる機能を追加しました。

新規SQL Server 2005接続設定

SQL Server 2005接続設定

新規SQL Server 2005接続設定

名前(N): SQL Server 2005接続設定(1)

種類(Y): SQL Server 2005接続設定

必須設定

ホスト名(S): localhost

ポート番号(Q): 1433

データベース名(D): sqlserver

☒ Windows統合認証を使用

ユーザ名(U): user

パスワード(P): ****

接続テスト(T)

完了(F) 取消し

グローバルリソースで設定を行います。Windows 統合認証を使用して接続する場合には、[Windows 統合認証を使用]にチェックを入れてください。(デフォルトはチェックなし)

また、別途ライブラリをインストールする必要があります。

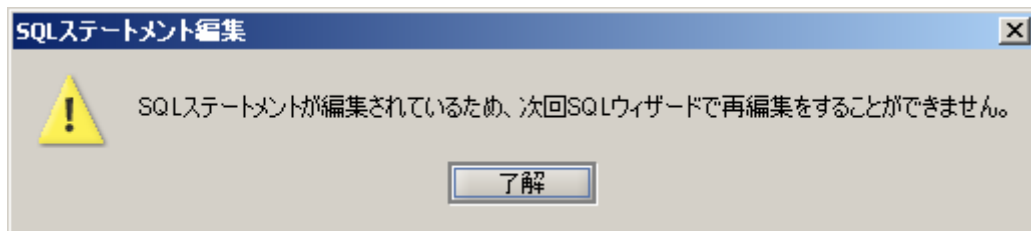
※詳細はヘルプ「ライブラリのインストール」を参照してください。

21. SQL ウィザード

21.1. SQL ステートメント再編集機能

SQL ウィザードにて作成した SQL ステートメントを、SQL ウィザードで再編集することができるようになりました。

SQL ウィザードにて作成された SQL ステートメントを手動で編集した場合、SQL ウィザードで再編集を行うことはできません。



SQL ウィザードの[SQL ステートメント]タブ内で SQL ステートメントを編集した場合は、上記の警告ダイアログが表示されます。



プロパティウィンドウの[SQL 文]で SQL ステートメントを編集した場合は、上記の警告ダイアログが表示されます。

22. Excel アダプタ

22.1. Excel2007 対応

Excel 2007 に対応しました。

22.2. 数式セルの値取得に関する仕様変更

シートから読み取り処理および定義済み範囲から読み取り処理で、数式セルの値の取得に関する仕様を変更しました。

- 表示形式が「数値」のセルの数式の結果値は、Excel の仕様と同様の有効桁数 15 桁の範囲の中で、書式に設定されている「小数点以下の桁数」まで読み取ります。
 - 有効桁数を超える桁の値は四捨五入します。
 - 設定されている「小数点以下の桁数」を超える小数点以下の桁の値は四捨五入します。
- 表示形式が「標準」のセルの数式の結果値は、Excel の仕様と同様の有効桁数 15 桁の範囲の中で、小数点以下は 9 桁まで読み取ります。
 - 有効桁数を超える桁の値は四捨五入します。
 - 整数部分が 0 ではない数値は、小数点 10 桁を四捨五入します。
 - 整数部分が 0 の数値は、0 以外の値が出現した小数点以下の桁から 9 桁を有効桁数とし、10 桁目を四捨五入します。

22.3. 書き込み処理

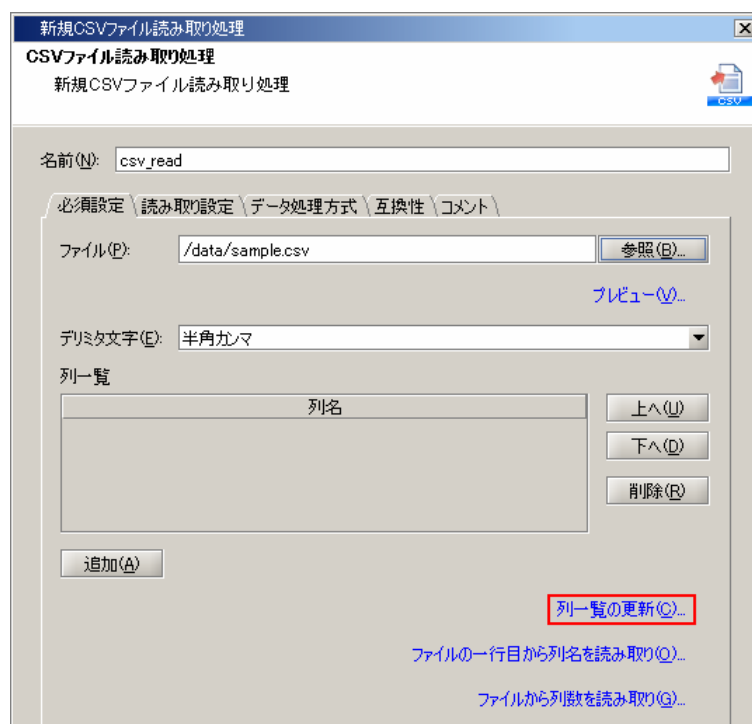
以下の設定がされているシートに対し、書き込みを行うことができるようになりました。

- クリップアート
- 「コントロール ツールボックス」のテキストボックス
- 図
- マクロ

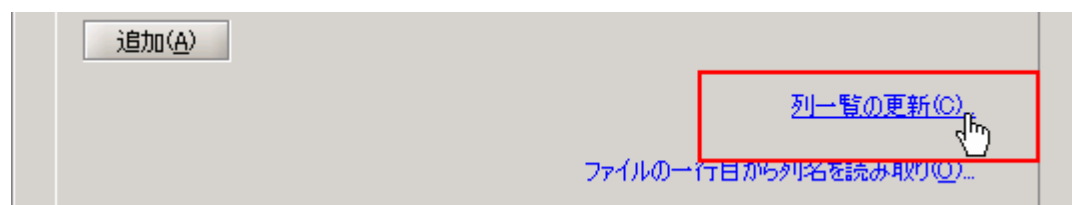
23. CSV アダプタ/Excel アダプタ

23.1. 列一覧の読み込み

列一覧を明示的に更新できる[列一覧の更新]リンクボタンを追加しました。従来まではファイルパスの入力時に自動的に列一覧を読み込んでいましたが、本機能の追加により、自動で列一覧で読み込むことはなくなります。



[列一覧の更新]リンクアクションを押下しないと[列一覧]に列は自動で設定はされません。



[ファイル]にファイルパスを入力または選択後、[列一覧の更新]リンクアクションを押下します。

列名	フィールド名
会社コード	
会社名	
担当者	
住所	
電話番号	

追加(A)

上へ(U)
下へ(D)
削除(R)

[列一覧]に列が定義されます。

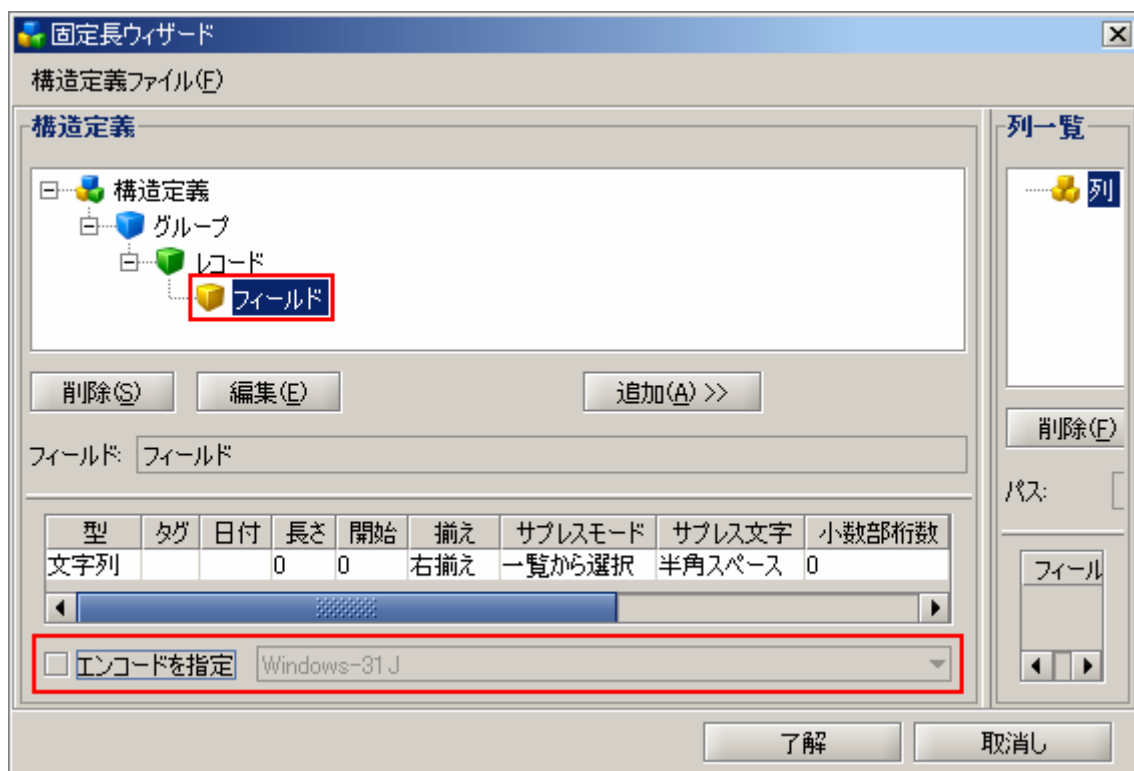
24. 固定長アダプタ

24.1. フィールドごとのエンコードの指定機能

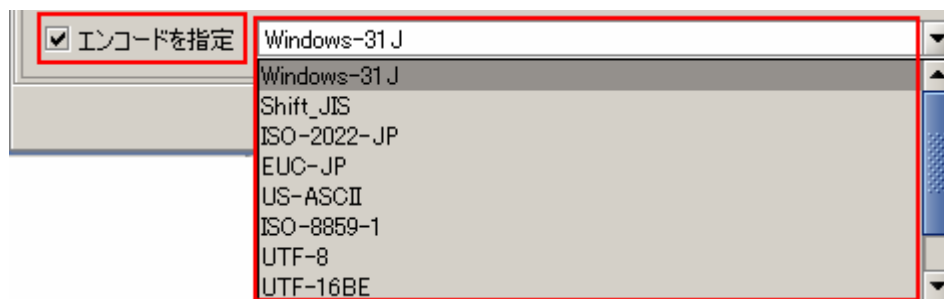
固定長ファイル読み取りおよび固定長ファイル書き込み処理で、各フィールドごとにエンコードを指定できるようにしました。

固定長ウィザードにて、フィールド選択時に[エンコードを指定]オプションにチェックを入れると、エンコードの選択が可能になります。

(※)デフォルトは「チェックなし」(指定しない)となります。その場合はプロパティ設定で指定したエンコードが使用されます。例えば固定長ファイル書き込み処理の場合の改行は、[書き込み設定]タブで指定したエンコードで書き込まれます。



フィールドの選択時に[エンコードを指定]プロパティが表示されます。



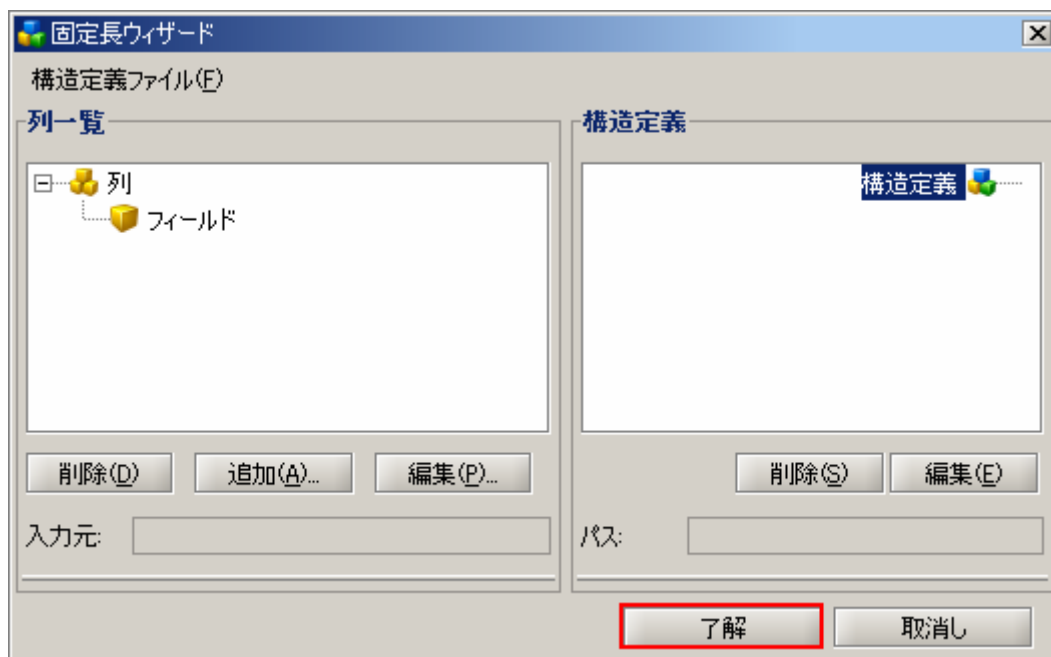
[エンコードを指定]にチェックを入れると、エンコード指定フィールドが有効になり、入力またはプルダウンメニューからエンコードの選択が可能になります。

25. 固定長/可変長ウィザード共通

25.1. デザイナでのフォーマット定義編集時の保存タイミングの変更

デザイナのリソース一覧編集画面から固定長・可変長ウィザードを起動してフォーマット定義を編集する際に、ウィザードの[了解]ボタンを押下すると、自動で保存が行われるようにしました。

2.2.1.1 までは「リソース一覧の編集」画面の[必須設定]タブから固定長/可変長ウィザードを起動してフォーマット定義を編集する場合には、ウィザードの[了解]ボタン押下後に、「リソース一覧の編集」画面の[適用]ボタンを押下する必要がありましたが、修正後はウィザードの[了解]ボタン押下時に保存されるようになります。



[了解]ボタン押下時に構造定義が保存されます。

26. Lotus Domino アダプタ

26.1. Lotus Domino 7.0 アダプタの追加

Lotus Domino 7.0 アダプタは、Lotus Domino 7.0 に接続が可能なアダプタです。

26.2. 添付ファイル名の自動変換

ビュー読み取り処理、ノーツ式読み取り処理および全文検索読み取り処理で、受信した添付ファイルに DataAnyware 2.0 ファイルシステムとして不正な文字が含まれていた場合に、「_」に変換するようにしました。

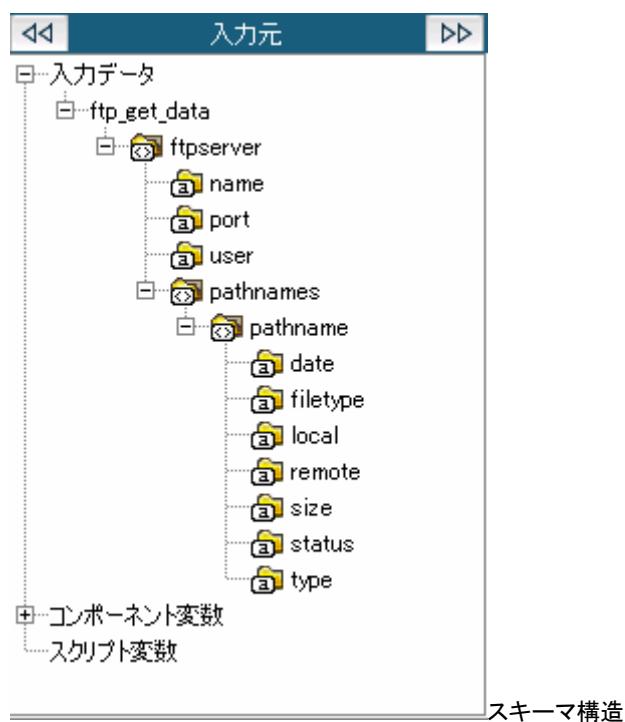
(※)DataAnyware 2.0 ファイルシステムの使用禁止文字については、Studio のオンラインヘルプを参照してください。

27. FTP アダプタ

27.1. 固定スキーマの自動設定

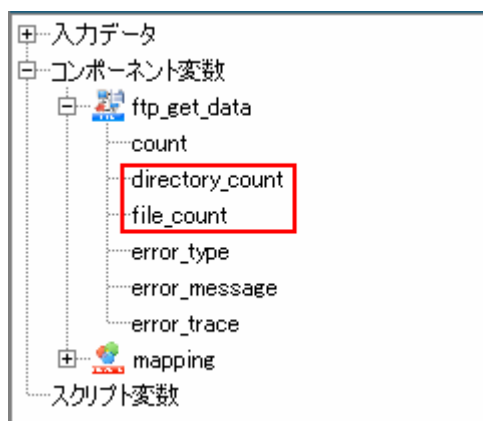
各オペレーションを Mapper の入力元とした際に、Mapper の入力元スキーマにスキーマが自動で表示されるようにしました。

また、スキーマの編集も「不可」となります。



27.2. コンポーネント変数の追加

取得したファイルの合計数が代入される「directory_count」およびディレクトリの合計数が代入される「file_count」の二つのコンポーネント変数を追加しました。



27.3. SFTP、FTP over SSL 対応

セキュア FTP(SFTP、FTP over SSL)接続に対応しました。

【プロパティ】

[セキュア FTP 接続設定]タブの[セキュリティタイプ]

【値の説明】

なし: 通常の FTP 接続を行います。

SFTP: SSH を使用した SFTP 接続を行います。

FTP over SSL: SSL を使用した FTP over SSL 接続を行います。

【デフォルト】

チェックなし

27.3.1.SFTP 接続

[秘密鍵]

公開鍵認証による接続を行いたいとき、秘密鍵ファイルのファイルパスを入力します。

[秘密鍵パスフレーズ]

秘密鍵のパスフレーズを入力します。

27.3.2.FTP over SSL 接続

必須設定 セキュアFTP接続設定

セキュリティタイプ(S): ☐ なし ☐ SFTP ☒ FTP over SSL

SSL接続タイプ(T): ☒ explicit ☐ implicit

クライアント証明書(F): 参照(B)...

証明書パスフレーズ(P):

☒ サーバ証明書のチェックを行わない

[SSL 接続タイプ]

explicit: explicit モードで接続を行います。

implicit: implicit モードで接続を行います。

[クライアント証明書]

クライアント証明書認証による接続を行いたいとき、クライアント証明書ファイル(PEM 形式)のファイルパスを入力します。

[証明書パスフレーズ]

クライアント証明書のパスフレーズを入力します。

[サーバ証明書のチェックを行わない]

サーバ証明書の妥当性チェックを行うかどうかを選択します。

チェックあり: サーバ証明書の妥当性チェックを行いません。(デフォルト)

チェックなし: サーバ証明書の妥当性チェックを行います。

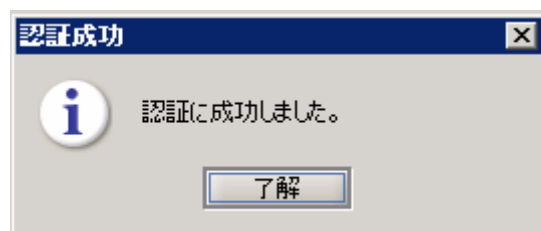
28. メールアダプタ

28.1. 認証テスト機能

複数メール送信処理および 1 通メール送信処理で、プロパティの[認証]タブの[認証モード]が[POP before SMTP]または[SMTP 認証(LOGIN)]の場合に、認証のテストを行うことができるプロパティアクション[認証テスト]を追加しました。

The screenshot shows a dialog box titled '新規複数メール送信処理' (New Multiple Mail Sending Process). Inside, there's a sub-header '複数メール送信処理' (Multiple Mail Sending Process) and '新規複数メール送信処理' (New Multiple Mail Sending Process). The '名前(N):' (Name) field contains 'mail_send_multiple_message'. The '入力データ(D):' (Input Data) dropdown is set to '(入力データなし)' (No input data). The '認証' (Authentication) tab is selected, showing '認証モード(A):' (Authentication Mode) with three radio buttons: 'なし(N)' (None), 'POP before SMTP' (selected), and 'SMTP 認証 (LOGIN)' (SMTP Authentication (LOGIN)). Below this are fields for 'ホスト名(S):' (Host Name) with 'test.mail.server', 'ポート番号(P):' (Port Number) with '110', 'ユーザ名(U):' (Username) with 'testuser', and 'パスワード(P):' (Password) with '*****'. A red box highlights the '認証テスト(T)' (Authentication Test) button at the bottom right.

認証テストに成功した場合、以下のダイアログが表示されます。



28.2. 差出人のメールアドレスのみを取得する（表示名は取得しない）機能

い）機能

メール受信(POP3)処理およびメール受信(IMAP4)処理で、プロパティの[受信設定]タブに差出人のメールアドレスのみを取得する(表示名は取得しない)[メールアドレスはアドレスのみ取得する]オプションを追加しました。

新規メール受信(POP3)処理

メール受信(POP3)処理

新規メール受信(POP3)処理

名前(N): mail_receive_pop3

必須設定 受信設定 添付ファイル ヘッダ データ処理方式 互換性 コメント

メール受信後(A): 削除しない

エンコード(E): 日本語自動判別(JIS Auto Detect)

☒ メールアドレスはアドレスのみ取得する(M)

デフォルトはチェックなしとなります。

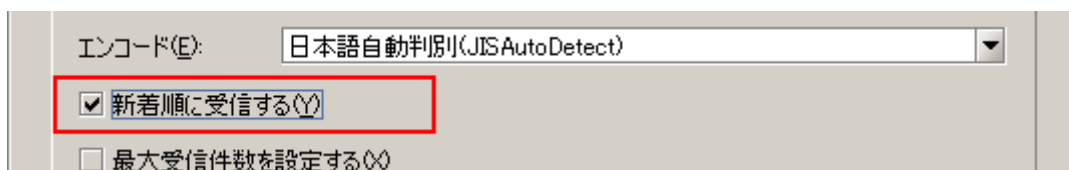
28.3. 添付ファイル名の自動変換

メール受信(POP3)処理およびメール受信(IMAP4)処理で、受信した添付ファイルに DataAnyware 2.0 ファイルシステムとして不正な文字が含まれていた場合に、「_」に変換するようにしました。

(※)DataAnyware 2.0 ファイルシステムの使用禁止文字については、Studio のオンラインヘルプを参照してください。

28.4. 新着順に受信する機能

メール受信(POP3)処理およびメール受信(IMAP4)処理で、新着順にメッセージを受信できる機能を追加しました。



【プロパティ】

[受信設定]タブの[新着順に受信する]

【値の説明】

チェックあり: 新しくメールサーバに届いた順にメッセージを受信します。

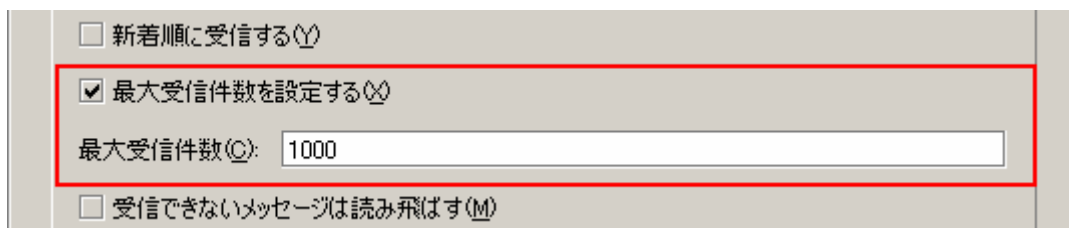
チェックなし: メールサーバに届いた順にメッセージを受信します。

【デフォルト】

チェックなし

28.5. 最大受信件数の指定機能

メール受信(POP3)処理およびメール受信(IMAP4)処理で、最大受信件数の指定ができる機能を追加しました。



【プロパティ】

[受信設定]タブの[最大受信件数を設定する]および[最大受信件数]

【値の説明】

チェックあり: [最大受信件数]に指定した件数を上限としてメッセージを受信します。([最大受信件数]のデフォルト値は「1000」です。)

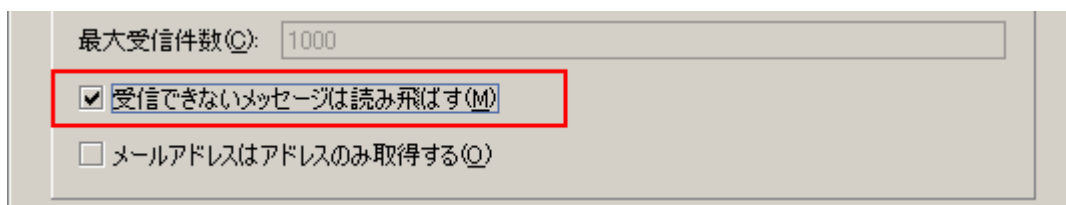
チェックなし: すべてのメッセージを受信します。

【デフォルト】

チェックなし

28.6. 受信できないメールをスキップする機能

メール受信(POP3)処理、メール受信(IMAP4)処理、受信できないメールをスキップして処理を継続することができる機能を追加しました。



最大受信件数(Q): 1000

☒ 受信できないメッセージは読み飛ばす(M)

☐ メールアドレスはアドレスのみ取得する(Q)

【プロパティ】

メール受信(POP3)処理、メール受信(IMAP4)処理
[受信設定]タブの[受信できないメールは読み飛ばす]

【値の説明】

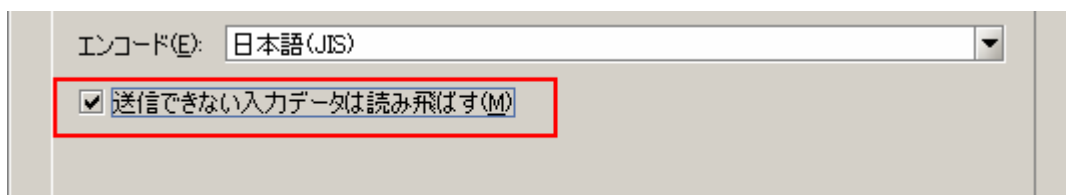
チェックあり: 受信できないメッセージはスキップして処理を継続します。
チェックなし: 受信できないメッセージを受信した際にはエラーにします。

【デフォルト】

チェックなし

28.7. 送信できない入力データをスキップする機能

複数メール送信処理で、送信できない入力データをスキップして処理を継続することができる機能を追加しました。



エンコード(E): 日本語(JIS)

☒ 送信できない入力データは読み飛ばす(M)

【プロパティ】

複数メール送信処理
[送信設定]タブの[送信できない入力データは読み飛ばす]

【値の説明】

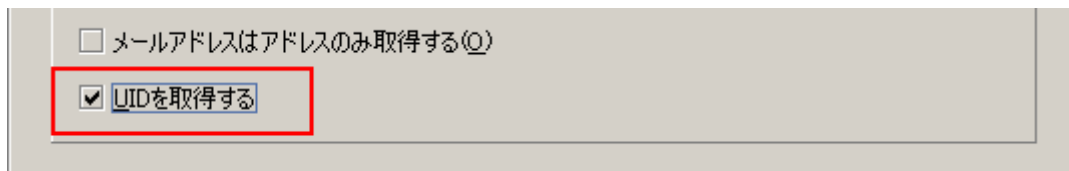
チェックあり: 送信できない入力データはスキップして処理を継続します。
チェックなし: 送信できないデータが入力された際にはエラーにします。

【デフォルト】

チェックなし

28.8. UID の取得機能

メール受信(IMAP4)処理で、UID の取得ができる機能を追加しました。



【プロパティ】

[受信設定]タブの[UID を取得する]

【値の説明】

チェックあり: UID を取得します。
チェックなし: UID を取得しません。

【デフォルト】

チェックなし

29. リポジトリ DB

29.1. リポジトリ DB とは

本製品では、ユーザ情報や連携処理、各種データを保管するリポジトリ領域を RDB(リレーショナルデータベース)に設定することができます。

このリポジトリ領域が設定された RDB を「リポジトリ DB」と呼びます。

リポジトリ DB は、DataAnyware 2.0 ファイルシステムやユーザ管理で使用されます。リポジトリ DB が設定されていないとユーザ管理の機能やファイルのアクセスコントロールの機能を使用することはできません。

(※) リポジトリ DB は必須ではありません。

(※) リポジトリ DB の詳細については、Studio のオンラインヘルプを参照してください。

29.2. リポジトリ DB の準備

本製品がリポジトリ DB としてサポートしているデータベースは以下の通りです。

- Oracle Database

動作確認済バージョン：11g R1・10g R2・10g R1・9i(9.2.0・9.0.1)・8i(8.1.7)

- DB2

動作確認済バージョン：V9.5・V9.1・V8.2・V7.2

- Microsoft SQL Server

動作確認済バージョン：2005・2000

- PostgreSQL

動作確認済バージョン：8.1・8.0

本製品は設定したデータベースに専用のテーブルを作成し、データを保存します。テーブルの作成は Server の初回起動時に行われます。すでに専用のテーブルが存在していた場合は新規に作成せず、既存のテーブルを使用します。



リポジトリ DB として使用するデータベースのインスタンスはリポジトリ DB 専用とし、他システムでの使用は行わないでください。

29.2.1.Oracle Database を使用する場合

Oracle Database をリポジトリ DB として使用するには、以下の情報が必要です。

- データベースサーバがインストールされているホストの IP アドレス
- サービスを提供しているポート番号(デフォルトは 1521)
- システム識別子(SID)
- Oracle に付属する JDBC ドライバへのパス
 - 11g R1 : [Oracle をインストールしたディレクトリ]/jdbc/lib/ojdbc5.jar
 - 10g R2 : [Oracle をインストールしたディレクトリ]/jdbc/lib/ojdbc14.jar
 - 10g R1、9.2.0 : [Oracle をインストールしたディレクトリ]/jdbc/lib/classes12.jar
 - 9.0.1、8.1.7 : [Oracle をインストールしたディレクトリ]/jdbc/lib/classes12.zip



10g R2 以外では、JDK1.4 用の JDBC ドライバ(ojdbc14.jar)は使用しないで下さい。

このとき、データベースに接続するための URL は

jdbc:oracle:thin:@<ホスト名>:<ポート番号>:<サービス名/SID>

となります。

データベースに新規のユーザを追加し、そのユーザに DBA のロールを設定してください。ここで設定した

- ユーザ ID
- パスワード

はインストール時に使用します。これらの値を用いて外部から接続できることをご確認ください。

29.2.2.DB2 を使用する場合

DB2 をリポジトリ DB として使用する場合には、以下の情報が必要です。

- DB2 サーバのホスト名もしくは IP アドレス
- サービスを提供しているポート番号
- データベース名
- DB2 に付属する JDBC ドライバへのパス
 - V9.5・V9.1・V8.2 : [DB2 をインストールしたディレクトリ]/sqlib/java/db2java.zip、db2jcc.jar
 - V7.2 : [DB2 をインストールしたディレクトリ]/sqlib/java/db2java.zip

このとき、DB2 に接続するための URL は

jdbc:db2://<ホスト名>:<ポート番号>/<データベース名>

となります(COM.ibm.db2.jdbc.net.DB2Driver を使用した接続をサポートしています)。

次に、データベースに新規のユーザを追加し、そのユーザに DBA のロールを設定してください。ここで設定した

- ユーザ ID
- パスワード

を使用して外部から接続できることをご確認ください。

29.2.3.Microsoft SQL Server を使用する場合

Microsoft SQL Server をリポジトリ DB として使用するには、以下の情報が必要です。

- 使用する SQL Server のホスト名もしくは IP アドレス
- サービスを提供しているポート番号
- SQL Server に付属する JDBC ドライバへのパス
 - 2005(Microsoft SQL Server 2005 JDBC Driver 1.2) : sqljdbc.jar
 - 2000(SQL Server 2000 Driver for JDBC Service Pack 3) : msbase.jar、mssqlserver.jar、msutil.jar

バージョン 2005 の場合の URL は、

jdbc:sqlserver://<ホスト名>:<ポート番号>;DatabaseName=<データベース名>

バージョン 2000 の場合の URL は、

jdbc:microsoft:sqlserver://<ホスト名>:<ポート番号>;DatabaseName=<データベース名>;SelectMethod=cursor

となります。

次に、データベースに新規のユーザを追加し、そのユーザに DBA のロールを設定してください。ここで設定した

- データベース名
- ユーザ ID
- パスワード

を使用して外部から接続できることをご確認ください

29.2.4. PostgreSQL を使用する場合

PostgreSQL をリポジトリ DB として使用する場合には、以下の情報が必要です。

- PostgreSQL サーバのホスト名
- サービスを提供しているポート番号
- データベース名
- ユーザ名/パスワード
- postgresql-8.X-XXX.jdbc3.jar へのパス
 - 8.1 : [PostgreSQL をインストールしたディレクトリ]/8.1/jdbc/ postgresql-8.X-XXX.jdbc3.jar
 - 8.0 : [PostgreSQL をインストールしたディレクトリ]/8.0/jdbc/ postgresql-8.X-XXX.jdbc3.jar

このとき、PostgreSQL に接続するための URL は

jdbc:postgresql://<ホスト名>:<ポート番号>/<データベース名>

となります。

次に、データベースに新規のユーザを追加し、そのユーザに DBA のロールを設定してください。ここで設定した

- ユーザ ID
- パスワード

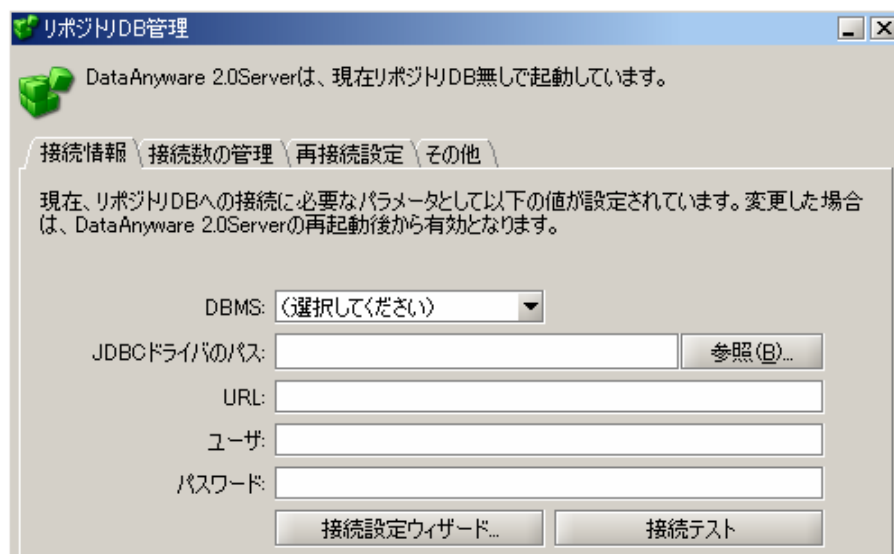
を使用して外部から接続できることをご確認ください

29.3. リポジトリ DB の適用設定

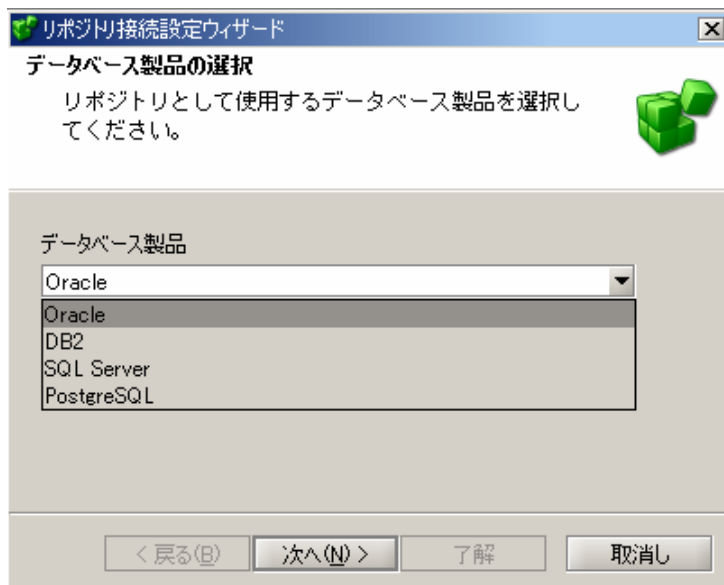
リポジトリ DB の適用は Studio 上から行います。

Studio にログインし、[コントロールパネル]-[リポジトリ DB 管理]

を選択してください。



表示されている[接続設定ウィザード]ボタンを押下してください。



リボジトリ DB に使用するデータベースの種類を選択し、

[次へ]ボタンを押下してください。



注意

リボジトリ DB 設定の後、Server を再起動するまで設定が有効になりません。

29.3.1.Oracle Database を使用する場合



使用するバージョンにあわせて表示されているファイルを選択し、

[次へ]ボタンを押下してください。

リポトリ接続設定ウィザード

データベース接続情報

リポトリとして使用するデータベースへの接続情報を入力してください。

ホスト名

ポート番号

SID

ユーザ名

パスワード

1521

< 戻る(B)

次へ(N) >

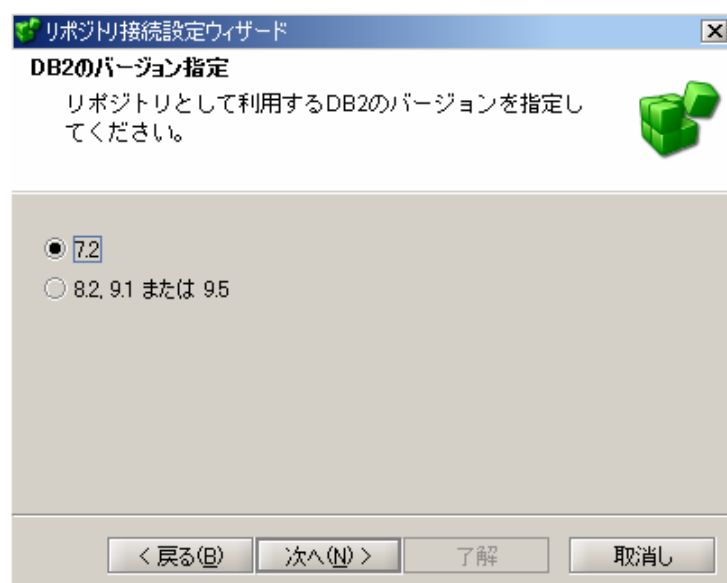
了解

取消し

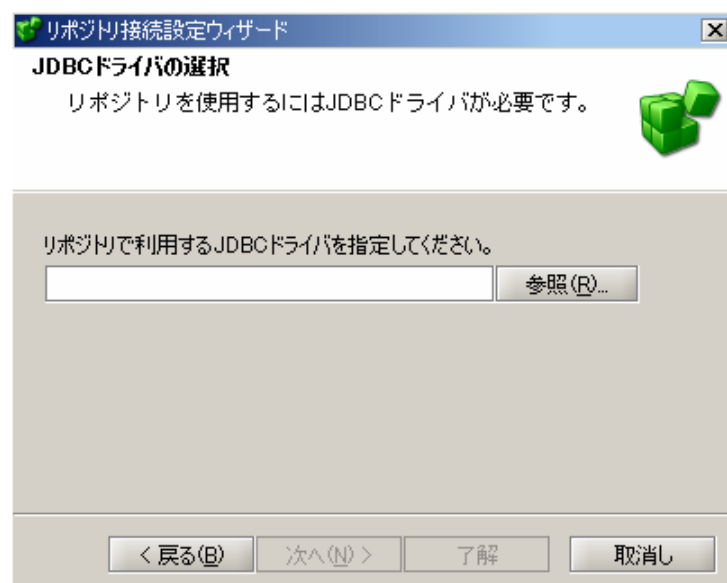
以下の情報を入力し、了解ボタンを押下してください。

項目	説明
ホスト名	Oracle がインストールされているホストの IP アドレス
ポート番号	サービスを提供しているポート番号(デフォルトは 1521)
SID	システム識別子(SID)
ユーザ名	データベースに接続するユーザ名
パスワード	ユーザのパスワード

29.3.2.DB2 を使用する場合



使用する DB2 のバージョンにチェックを入れて[次へ]ボタンを押下してください。デフォルトでは[7.2]がチェックされています。



使用するバージョンにあわせて表示されているファイルを選択し、
[次へ]ボタンを押下してください。

- V9.5・V9.1・V8.2 : [DB2 をインストールしたディレクトリ]/sqlib/java/db2java.zip、db2jcc.jar
- V7.2 : [DB2 をインストールしたディレクトリ]/sqlib/java/db2java.zip

リポジトリ接続設定ウィザード

データベース接続情報

リポジトリとして使用するデータベースへの接続情報
を入力してください。

ホスト名

ポート番号

データベース名

ユーザ名

パスワード

6789

< 戻る(B)

次へ(N) >

了解

取消し

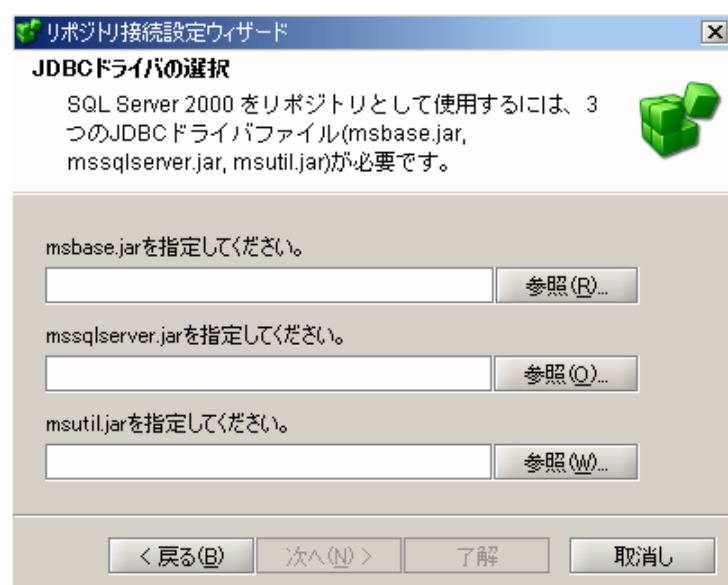
以下の情報を入力し、了解ボタンを押下してください。

項目	説明
ホスト名	DB2 がインストールされているホストの IP アドレス
ポート番号	サービスを提供しているポート番号
データベース名	使用するデータベース名
ユーザ名	データベースに接続するユーザ名
パスワード	ユーザのパスワード

29.3.3. Microsoft SQL Server を使用する場合



使用する SQL Server のバージョンにチェックを入れて[次へ]ボタンを押下してください。デフォルトでは[2000]がチェックされています。



使用するバージョンにあわせて表示されているファイルを選択し、

[次へ]ボタンを押下してください。

- 2005(Microsoft SQL Server 2005 JDBC Driver 1.2) : sqljdbc.jar
- 2000(SQL Server 2000 Driver for JDBC Service Pack 3) : msbase.jar、mssqlserver.jar、msutil.jar

リポジトリ接続設定ウィザード

データベース接続情報

リポジトリとして使用するデータベースへの接続情報を入力してください。

ホスト名

ポート番号

データベース名

ユーザ名

パスワード

1433

< 戻る(B)

次へ(N) >

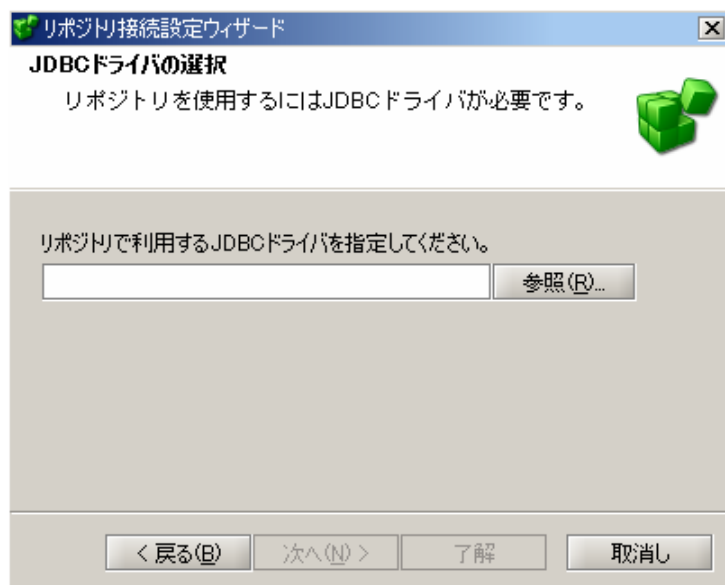
了解

取消し

以下の情報を入力し、了解ボタンを押下してください。

項目	説明
ホスト名	SQLServerがインストールされているホストのIPアドレス
ポート番号	サービスを提供しているポート番号(デフォルトは1433)
データベース名	使用するデータベース名
ユーザ名	データベースに接続するユーザ名
パスワード	ユーザのパスワード

29.3.4. PostgreSQL を使用する場合



使用するバージョンにあわせて表示されているファイルを選択し、

[次へ]ボタンを押下してください。

- 8.1 : [PostgreSQL をインストールしたディレクトリ]/8.1/jdbc/ postgresql-8.X-XXX.jdbc3.jar
- 8.0 : [PostgreSQL をインストールしたディレクトリ]/8.0/jdbc/ postgresql-8.X-XXX.jdbc3.jar

リポジトリ接続設定ウィザード

データベース接続情報

リポジトリとして使用するデータベースへの接続情報を入力してください。

ホスト名

ポート番号

データベース名

ユーザ名

パスワード

5432

< 戻る(B)

次へ(N) >

了解

取消し

以下の情報を入力し、了解ボタンを押下してください。

項目	説明
ホスト名	PstgreSQL がインストールされているホストの IP アドレス
ポート番号	サービスを提供しているポート番号(デフォルトは 5432)
データベース名	使用するデータベース名
ユーザ名	データベースに接続するユーザ名
パスワード	ユーザのパスワード

新機能ガイド

第1版 2010 年 01 月 08 日

発行 JB アドバンスド・テクノロジー株式会社

〒221-0022 神奈川県横浜市神奈川区守屋町三丁目 9 番地 C 号棟

お問い合わせ 弊社ホームページより、お問い合わせください。
<http://www.jbat.co.jp/>

本書は著作権上の保護を受けており、本書の全部あるいは一部に関して、JB アドバンスド・テクノロジー株式会社からの文書による許諾を得ず、無断で複写、複製することは禁じられています。また、本書はユーザーへ通知することなく変更される場合があります。

