

Wolf DB Manager RCP 2



1 はじめに	4
1-1 対象データベース製品.....	4
1-2 ドライバの準備.....	5
2 起動と初期設定	6
2-1 起動方法.....	6
2-2 DB 接続初期設定.....	7
2-2-1 JDBCドライバ追加手順.....	7
2-2-2 接続設定追加手順.....	8
2-2-3 RDB への接続設定サンプル (PostgreSQL, Oracle).....	10
2-2-4 Access への接続設定サンプル.....	11
2-2-5 FileMaker への接続設定サンプル.....	12
2-3 SSHトンネリング設定.....	13
2-3-1 SSHトンネリング定義の追加と利用設定手順.....	13
3 各部の名称と説明	15
3-1 名称と説明.....	15
3-2 ツールチップ (TOOL TIP)(アイコンの説明).....	15
4 SQL エディタとSQL 文の実行	16
4-1 SQL エディタの表示方法.....	16
4-1-1 SQL アイコンからの表示.....	16
4-1-2 Start ボタンからの表示.....	16
4-1-3 DB Explorer からの表示.....	17
4-1-4 Navigation ビューからの表示.....	18
4-2 SQL エディタの操作方法.....	19
4-2-1 ツールバー.....	19
4-2-2 SQL 履歴ビュー.....	19
4-3 SQL エディタによるSQL 文の実行.....	20
4-4 SQL エディタによる実行計画の取得.....	20
4-5 SQL 文の整形(フォーマット).....	21
4-5-1 整形実行.....	21
4-5-2 整形設定.....	21
5 LST エディタとSQL 文の実行	22
5-1 LST エディタの表示方法.....	22
5-1-1 LST アイコンからのLST エディタの新規表示.....	22
5-1-2 Start ボタンからの表示.....	22
5-1-3 Navigation ビューからの表示.....	23
5-2 LST エディタの操作方法.....	24
5-2-1 ツールバーとアウトライン.....	24
5-2-2 レイアウト調整.....	25
5-3 LST エディタによるSQL 文の実行.....	26
5-3-1 発行するSQL 文の確認方法.....	26
5-3-2 データ抽出条件、表示順、重複除去の指定方法.....	27
5-3-3 抽出結果に依存したデータ抽出.....	28

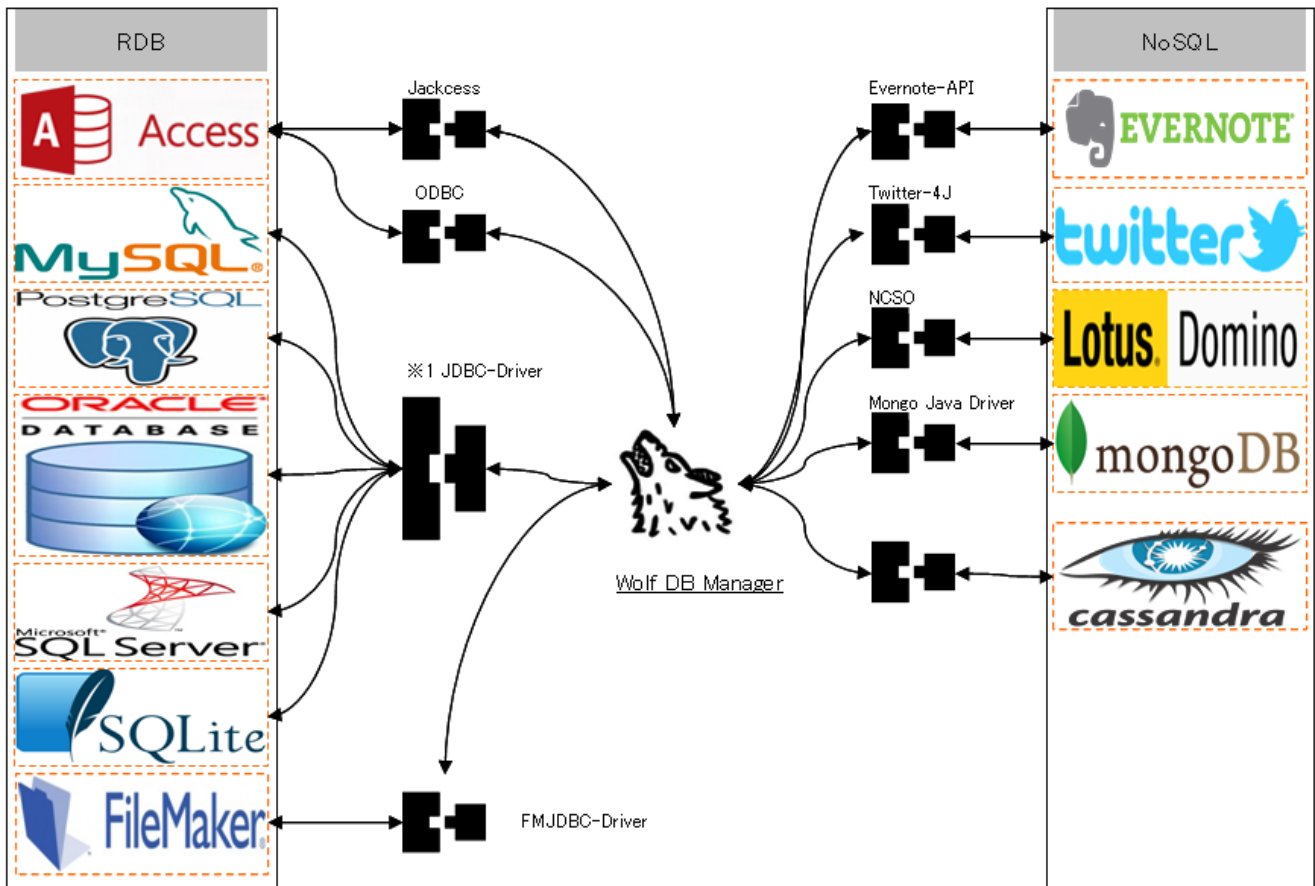
6 逆引き	29
6-1 任意のテーブルを対象にレコード数の増減を確認したい.....	29
6-2 データの変更内容を確認したい、エビデンスとして残したい.....	31
6-3 データを別のデータベースへ移行したい.....	34
6-4 新しいバージョンの WOLF に古いバージョンの設定を引継ぎたい.....	35
7 テストシナリオランナー	36
7-1 概要.....	36
7-2 操作方法.....	37
7-2-1 起動方法.....	37
7-3 シナリオ実行初期設定.....	38
7-3-1 設定.....	38
7-4 シナリオの自動操作に関する記述.....	39
7-4-1 (主にブラウザへの) 命令.....	39
7-4-2 パラメータ (変数) の展開.....	40
7-5 自動操作に起因するデータベース検証処理の記述 (任意).....	41
7-5-1 検証アクション.....	41
7-5-2 <i>select_query</i> (<i>SELECT</i> 結果の確認) アクション.....	42
7-6 自動操作に起因するブラウザ検証処理の記述 (任意).....	43
7-6-1 ブラウザの内容変化に伴う検証.....	43
8 FAQ	44
8-1 起動しません.....	44
8-1-1 JavaVM が未インストール状態により起動しない.....	44
8-1-2 実行権限が無いので起動しない.....	45
8-1-3 32bitOS の PC にて 64bit 用のアプリケーションを実行することで起動しない.....	46
8-1-4 利用方法をまとめたページは他にありませんか?.....	46

1 はじめに

Wolf DB Manager RCP は「Eclipse Rich Client Platform」で開発した DB 用統合開発環境ソフトウェアです。

1-1 対象データベース製品

MySQL、PostgreSQL、Oracle、SQLServer、SQLite、Access、twitter、Domino、mongoDB、Evernote での御利用が可能です。



※1 JDBC-Driver ファイルは『1-2 ドライバの準備』の URL から御利用製品に応じたファイルをダウンロードしてください。

1-2 ドライバの準備

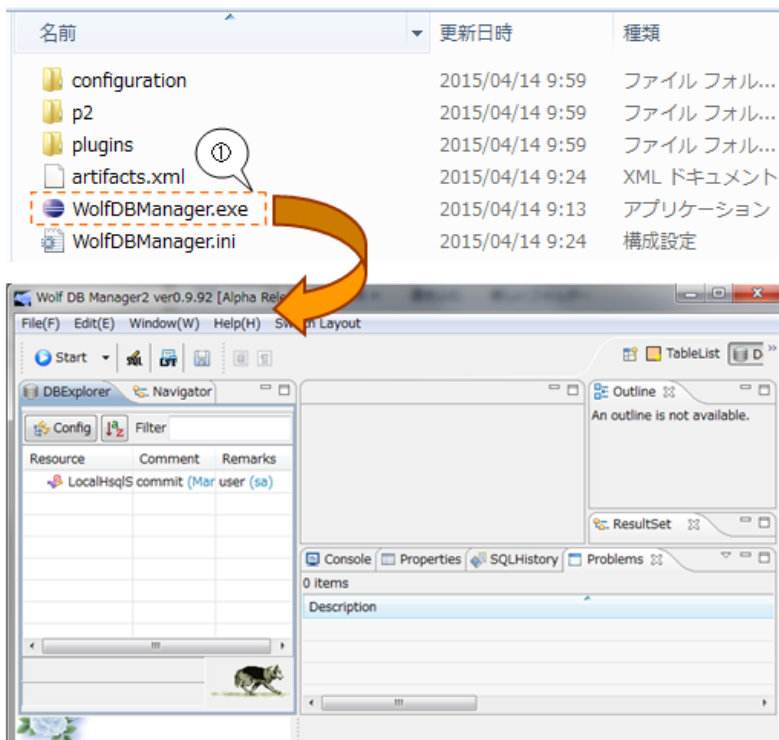
御利用 DB に応じたドライバを御準備いただきます。

No	DB Vendor	JDBC Driver File	JDBC Driver Class
	Driver Download URL		
1	MySQL	mysql-connector-java-xxx.jar	com.mysql.jdbc.Driver
	http://dev.mysql.com/downloads/connector/j/		
2	PostgreSQL	postgresql-xxx.jdbc4.jar	*postgresql.*
	https://jdbc.postgresql.org/download.html		
3	Oracle	ojdbc14.jar	oracle.jdbc.driver.OracleDriver
	http://www.oracle.com/technetwork/jp/database/enterprise-edition/jdbc-111060-097832-ja.html		
4	MS-SQL Server	sqljdbc4.jar	*sqlserver.*
	http://www.microsoft.com/ja-jp/download/details.aspx?id=11774		
5	SQLite	sqlite-jdbc-xxx.jar	org.sqlite.JDBC
	https://bitbucket.org/xerial/sqlite-jdbc/downloads		
6	MS-ACCESS	不要(組込済)	
7	Twitter	不要(組込済)	twitter4j.Twitter
8	IBM Lotus Domino Server	NCSO.jar	lotus.domino.database
9	Mongo-db	不要(組込済)	com.mongodb.Mongo
10	Cassandra	不要(組込済)	

2 起動と初期設定

2-1 起動方法

- ① ダウンロードファイルを解凍後、『WolfDBManager.exe』をダブルクリックすると起動します。



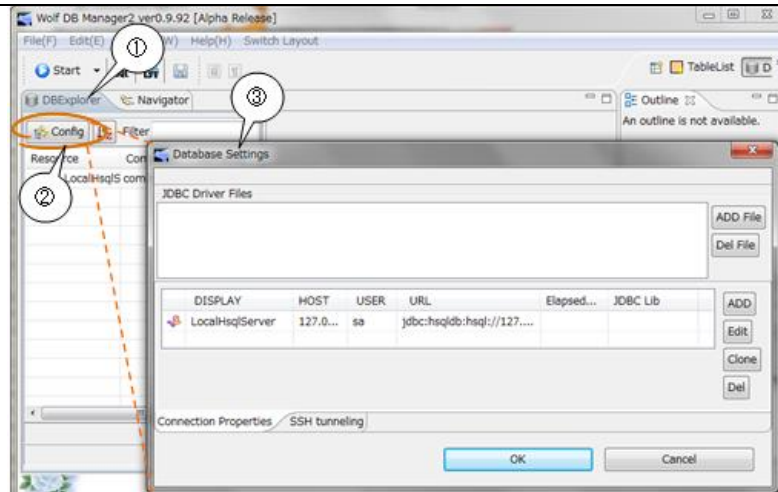
① One Point 起動しない場合

表示されたメッセージに応じた FAQ を御確認ください

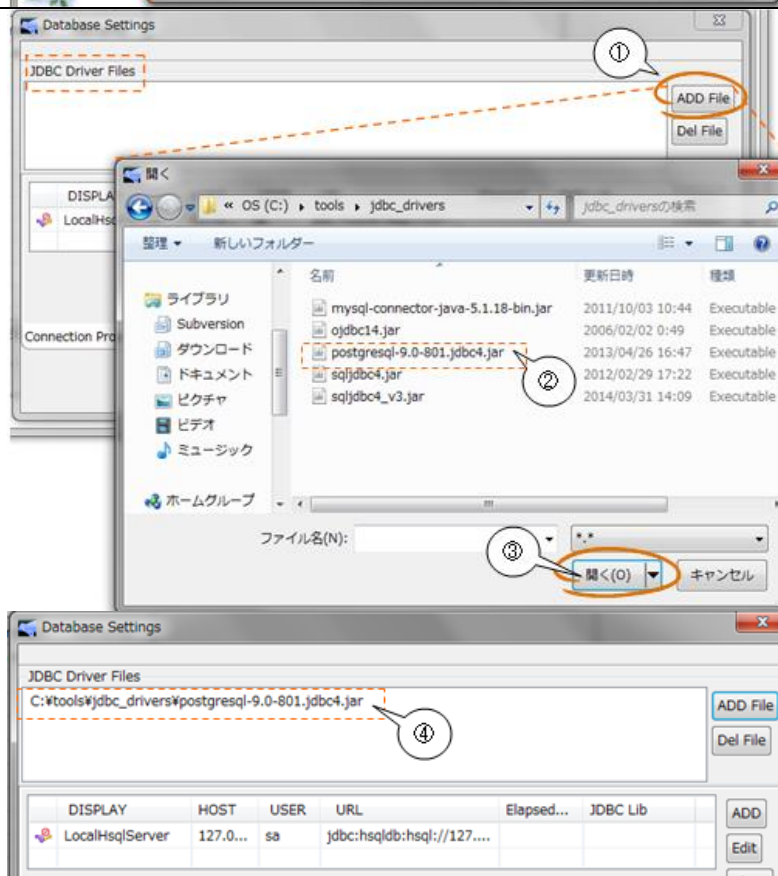
2-2 DB 接続初期設定

接続定義は 2 段階で行います。1 段階目は利用するデータベース製品に応じた JDBC ドライバファイルの登録。2 段階目は前述製品内へ作成したデータベースへの接続設定情報の登録となります。
同じデータベース製品への接続設定を増やす際は、1 段階目の作業は不要となります。

2-2-1 JDBC ドライバ追加手順



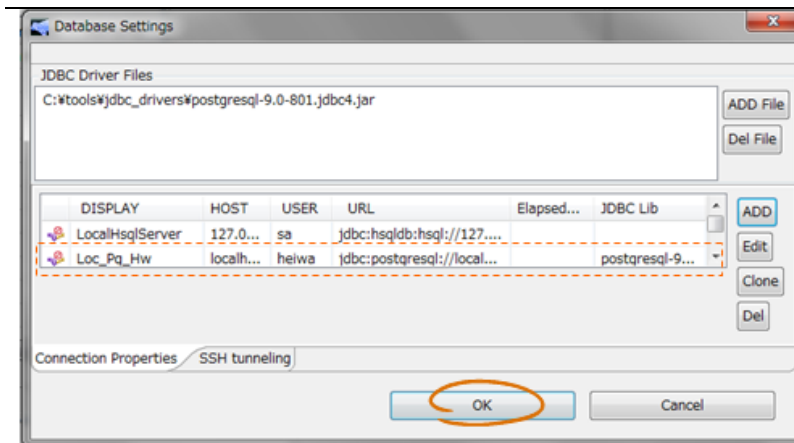
- 1 ①『DB Explorer』ビューの②『Config』ボタンを押下し、③『Database Setting』ダイアログを表示する



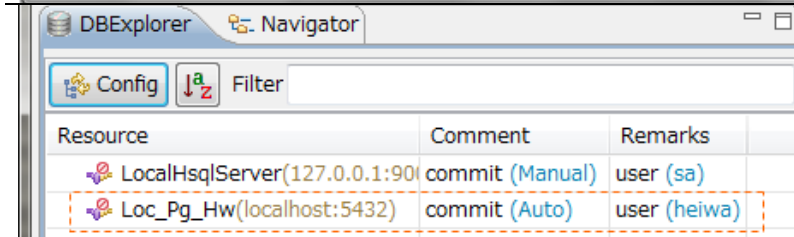
- 2 JDBC ドライバの登録
①『ADD File』ボタンを押下し、接続先の DB に対応した JDBC Driver ファイルを選択する子画面を開く
②JDBC ドライバファイルを指定し、③確定する
④JDBC Driver ファイルリストに上記で指定したファイルがリストアップされます

2-2-2 接続設定追加手順

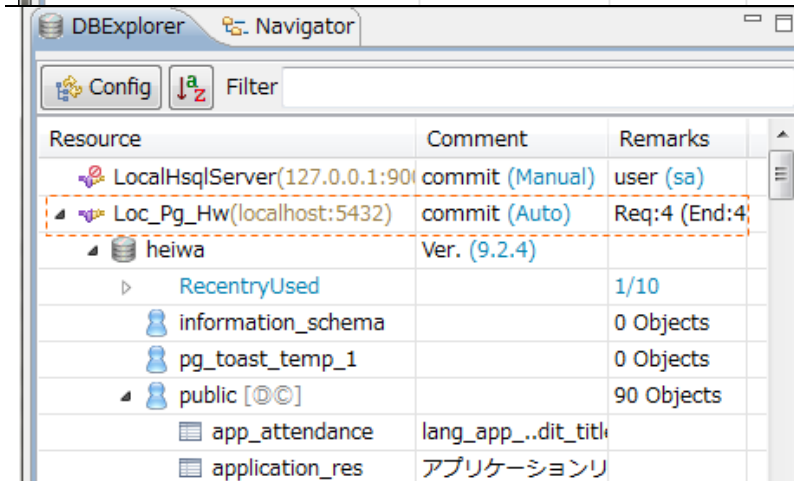
	<p>1 ①『ADD』ボタンを押下し、②設定ダイアログを表示する</p>
	<p>2 JDBCドライバクラスの指定 ①ドライバクラス選択プルダウンにて、②利用するクラスを指定する</p>
	<p>3 基本設定情報の指定</p> <p>①接続定義名称の指定 例)「Loc_Pg_Hw」 ②接続先 URL の指定 ③ユーザ ID の指定 ④パスワードの指定</p> <p>上記設定後、以下を行う</p> <p>⑤接続テストの実施 ⑥結果確認ダイアログの「ok」ボタン押下 ⑦設定情報の一時保存</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>①One Point 2-2-3 参照</p> </div>



4 設定情報の保存



5 『DB Explorer』ビューに#3 にて定義した情報が追加されていることを確認する



6 ダブルクリックし、DB に接続し、DB の定義が取得できることを確認する

2-2-3 RDB への接続設定サンプル (PostgreSQL, Oracle)

Java-Web アプリケーションの DB 接続定義は jdbc.properties に記述されることが多いことから、それを例に接続設定のサンプルを提示します。

●PostgreSQL 且つ Spring-JDBC

定義内容	設定サンプル
<pre> jdbc.driverClassName=org.postgresql.Driver jdbc.url=jdbc:postgresql://127.0.0.1:5432/caffe_latte jdbc.username=asus jdbc.password=mt_rainier </pre>	

●Oracle 且つ Spring-JDBC

定義内容	設定サンプル
<pre> jdbc.driverClassName=oracle.jdbc.driver.OracleDriver jdbc.url=jdbc:oracle:thin:@127.0.0.1:1521:xe jdbc.username=caramel jdbc.password=nuts </pre>	<p>[1]JDBC Driver Class にて『Oracle』が選択できない場合は、2-2-1#2 にて『ojdbc14.jar』の追加が漏れています。</p>

2-2-4 Access への接続設定サンプル

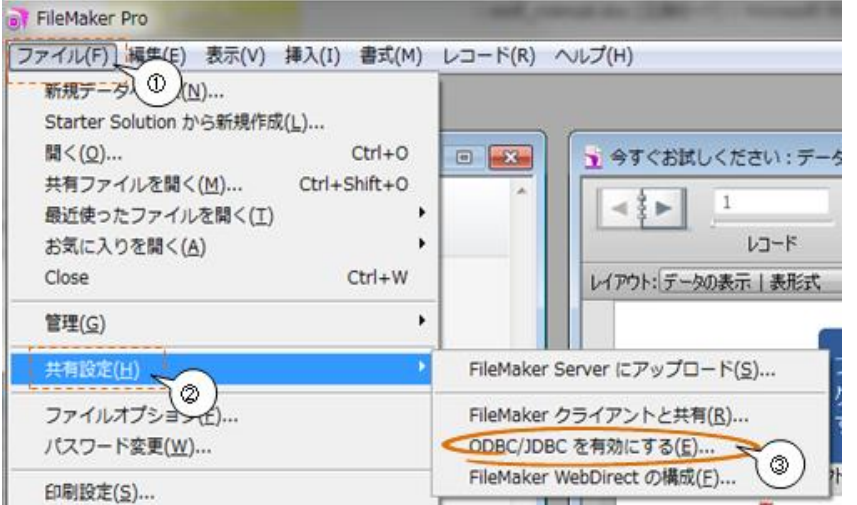
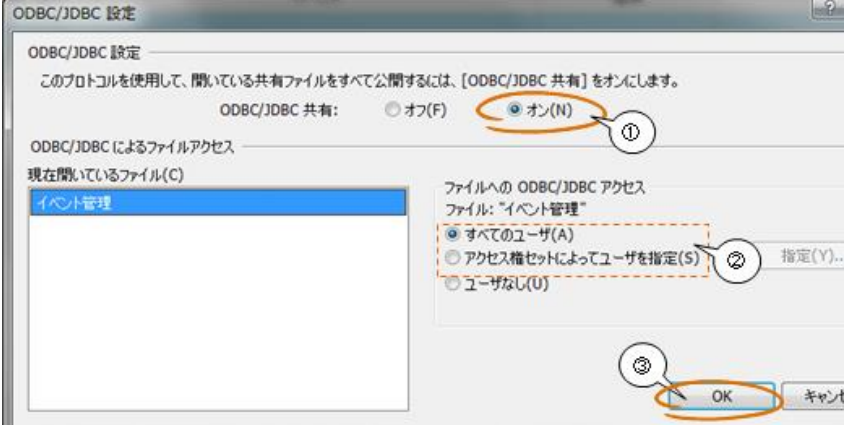
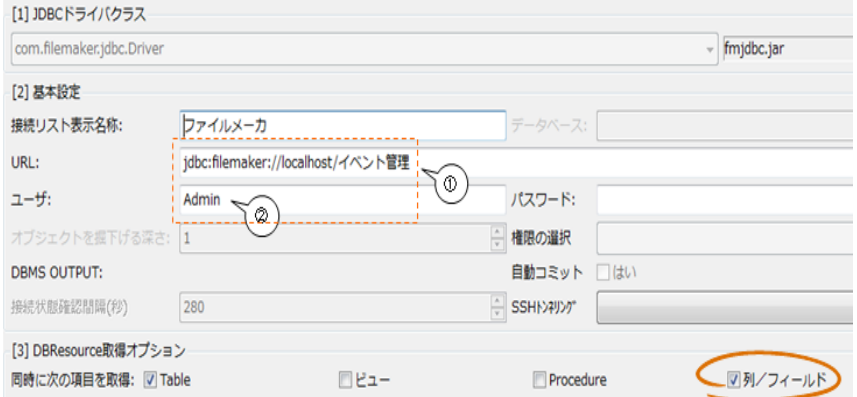
●ACCESS の MDB ファイルへ JACKCESS を利用して接続

	設定サンプル												
<p>①ドライバリストのプルダウンを開く</p> <p>②Jackcess を選択</p> <p>③対象となる mdb を選択</p> <p>④『開く』を指定</p> <p>⑤表示名称を指定</p> <p>⑥、⑦変更しない</p>	<p>[1] JDBC Driver Class</p> <p>org.postgresql.Driver oracle.jdbc.driver.OracleDriver org.hsqldb.jdbcDriver sun.jdbc.odbc.JdbcOdbcDriver com.mongodb.Driver twitter4j.TwitterDriver evernote. MS Access with Jackcess DBMS OUTPUT: Auto Commit <input type="checkbox"/> Yes</p> <p>開く</p> <p>整理 新しいフォルダー</p> <table border="1"> <thead> <tr> <th>名前</th> <th>更新日時</th> <th>種類</th> </tr> </thead> <tbody> <tr> <td>dwg_caffe.mdb</td> <td>2015/03/03 8:54</td> <td>Microsoft Office Access 2010 データベース</td> </tr> <tr> <td>dwg_latte.mdb</td> <td>2015/02/18 15:16</td> <td>Microsoft Office Access 2010 データベース</td> </tr> <tr> <td>dwg_nuts.mdb</td> <td>2015/03/02 20:37</td> <td>Microsoft Office Access 2010 データベース</td> </tr> </tbody> </table> <p>ファイル名(N): dwg_latte.mdb Microsoft ACCESS Database</p> <p>開く(O) キャンセル</p> <p>[1] JDBC Driver Class</p> <p>MS Access with Jackcess</p> <p>[2] Basic Properties</p> <p>DISPLAY NAME: dwg_latte.mdb DATABASE:</p> <p>MDB: C:\tmp\drawing_import_top\dwg_latte.mdb</p> <p>USER: guest PASSWORD:</p>	名前	更新日時	種類	dwg_caffe.mdb	2015/03/03 8:54	Microsoft Office Access 2010 データベース	dwg_latte.mdb	2015/02/18 15:16	Microsoft Office Access 2010 データベース	dwg_nuts.mdb	2015/03/02 20:37	Microsoft Office Access 2010 データベース
名前	更新日時	種類											
dwg_caffe.mdb	2015/03/03 8:54	Microsoft Office Access 2010 データベース											
dwg_latte.mdb	2015/02/18 15:16	Microsoft Office Access 2010 データベース											
dwg_nuts.mdb	2015/03/02 20:37	Microsoft Office Access 2010 データベース											

2-2-5 FileMaker への接続設定サンプル

【参考サイト】 <http://www.filemaker.com/help/13/fmp/ja/html/odbc.19.2.html>

●FileMaker の FMP ファイルへ FMJDBC ドライバを利用して接続

	設定サンプル
<p>①『ファイル』メニューを選択 ②『共有設定』を選択 ③『ODBC/JDBC を有効にする』を選択</p>	
<p>①『オン』を選択 ②『すべてのユーザ』または『アクセス権セットによってユーザを指定』を選択 ③『OK』を指定</p>	
<p>①接続したい FMP ファイル名を入力する ②Admin または上記の②で指定したユーザ</p>	

2-3 SSHトンネリング設定

DB 接続ポートへの接続許可がローカルサーバーのみで、該当サーバーへは SSH のポートしか接続許可されていない場合、SSH の接続を介して DB に繋げることが可能です。

putty や teraterm のトンネリング機能を使わずに Wolf だけで行う場合は以下の SSH 接続定義を行い、2-2 で作成した DB 定義から SSH 接続定義を参照するよう設定します。

2-3-1 SSHトンネリング定義の追加と利用設定手順

	<p>1 『DB Explorer』ビューの『Config』ボタン押下し、『Database Setting』ダイアログを表示する</p>
	<p>2 ①『SSH トンネリング』タブを表示し、②『追加』ボタンを押下し、設定ダイアログを開く</p> <p>③SSH 接続定義名称を指定 ④接続先サーバーアドレスを指定 ⑤上記の SSH ポートを指定 ⑥SSH ログインユーザ ID を指定 ⑦SSH ログインパスワードを指定 ⑧ローカル PC の空きポートを指定 ⑨接続先サーバーのDBの待ち受けポートを指定</p>
	<p>3 定義されたことを確認</p>
	<p>4 SSHトンネリングするDB 接続定義を指定し、編集ボタンを押下</p>

[1] JDBC Driver Class
org.postgresql.Driver postgresql-9.0-801.jdbc4.jar

[2] Basic Properties

DISPLAY NAME: acer_t DATABASE:

URL: jdbc:postgresql://localhost:15432/acer_t_db

USER: pos_user PASSWORD: ●●●●●●●●

RECURSIVE SEARCH DEPTH: 1 SYSDBA or SYSOPER or Nothing

DBMS OUTPUT: Auto Commit Yes

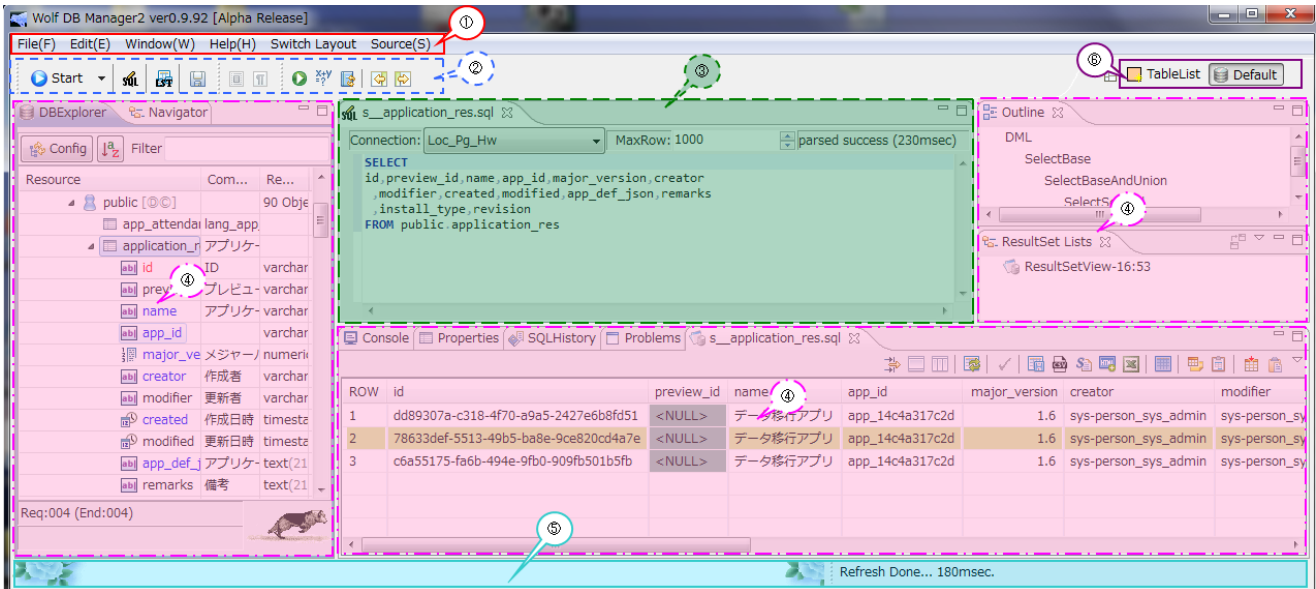
Keep alive interval(sec): 280 SSH tunneling g_dev_server

5 接続先 URL 中のポート番号が#2 の⑧ローカル PC の空きポート番号と同じこと

『SSH トネリング』の選択値が#2 の③の接続定義名称であること

3 各部の名称と説明

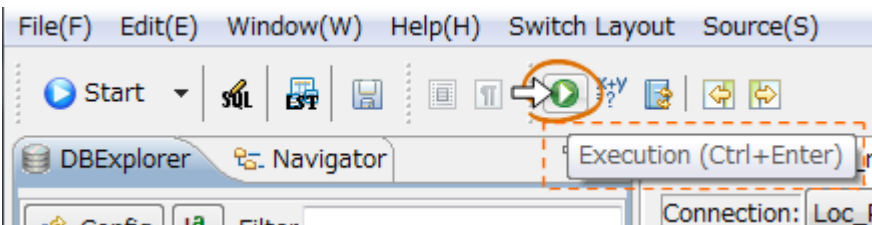
3-1 名称と説明



NO	名称	説明	操作方法参照先
1	メニューバー	全ての操作を行うメニューが格納されています	
2	ツールバー	ショートカットメニューがアイコンで表示されます	
3	エディタ	SQL 文を記述修正する SQL エディタや MS-ACCESS のように GUI 上でテーブルの関連、検索、表示順を指定する LST エディタを表示します	
4	ビュー	各種情報を表示します	
5	ステータスバー	処理中、処理後のメッセージを表示します	
6	パースペクティブ切替	SQL 文の発行用パースペクティブとテーブルリスト用パースペクティブを切替ます	

3-2 ツールチップ (Tool Tip) (アイコンの説明)

アイコンの意味はマウスカースルを数秒合わせることで表示されます。



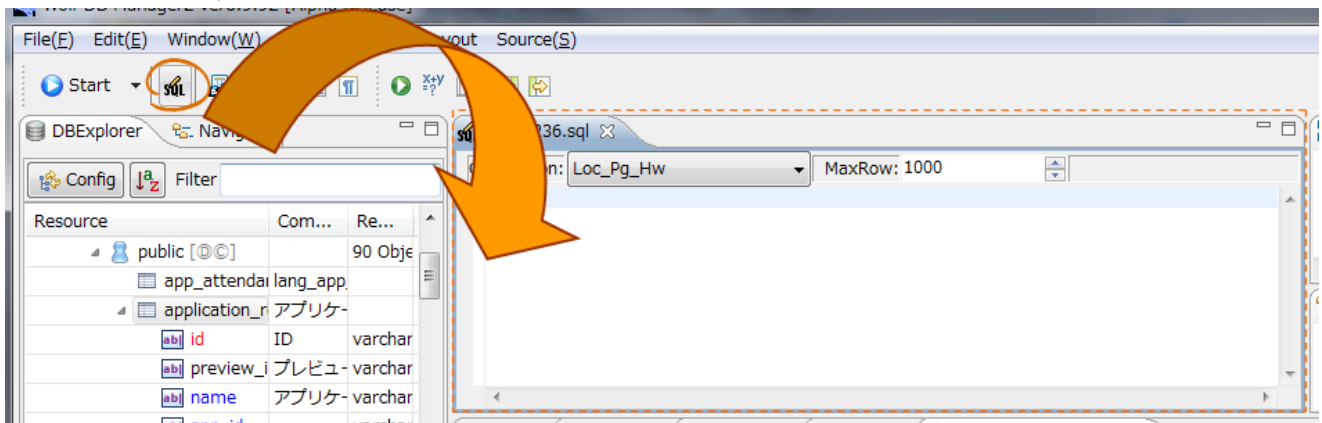
4 SQL エディタと SQL 文の実行

4-1 SQL エディタの表示方法

SQL エディタは 4 つの方法で表示することが可能です。

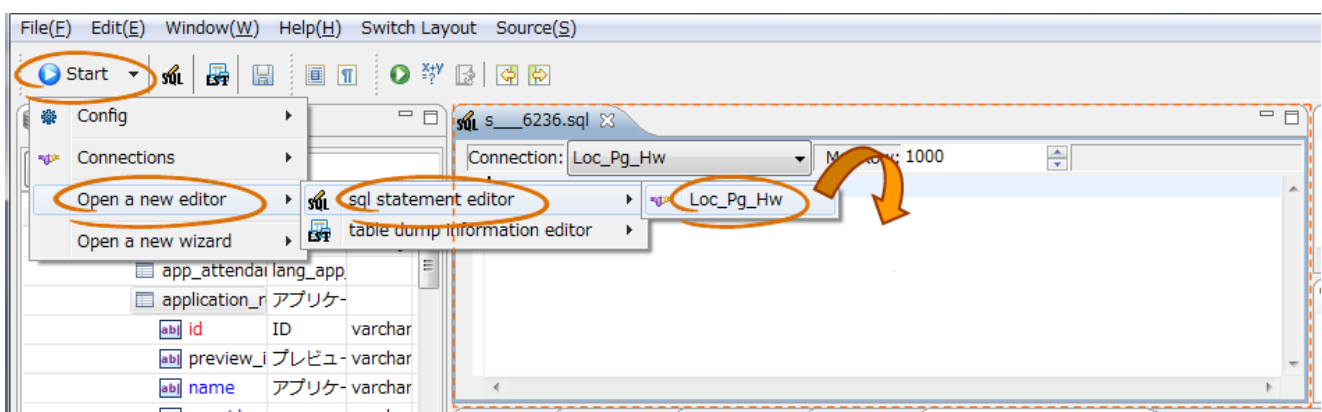
4-1-1 SQL アイコンからの表示

ツールバーの『SQL』アイコンをクリックする。



4-1-2 Start ボタンからの表示

ツールバーの『Start』 - 『Open a new editor』 - 『sql statement editor』 - 『接続済コネクション名』をクリックする。

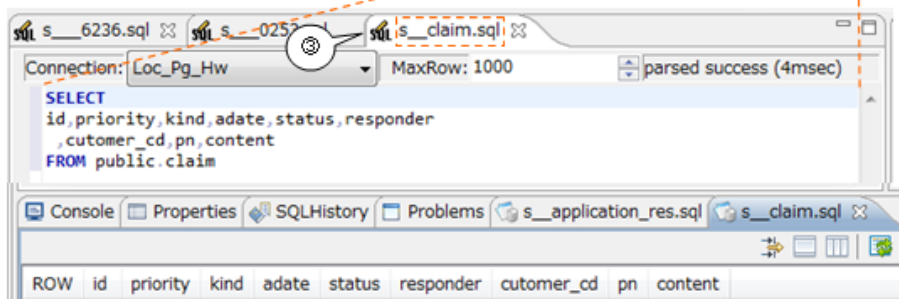
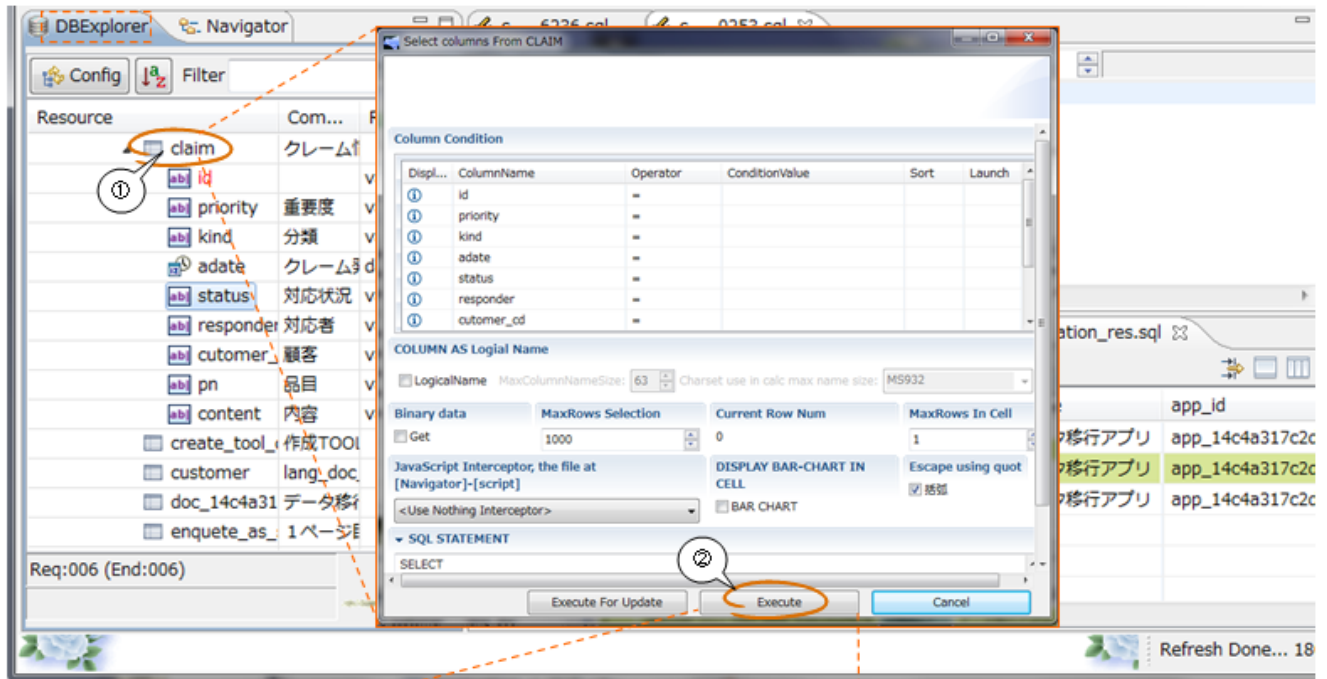


4-1-3 DB Explorer からの表示

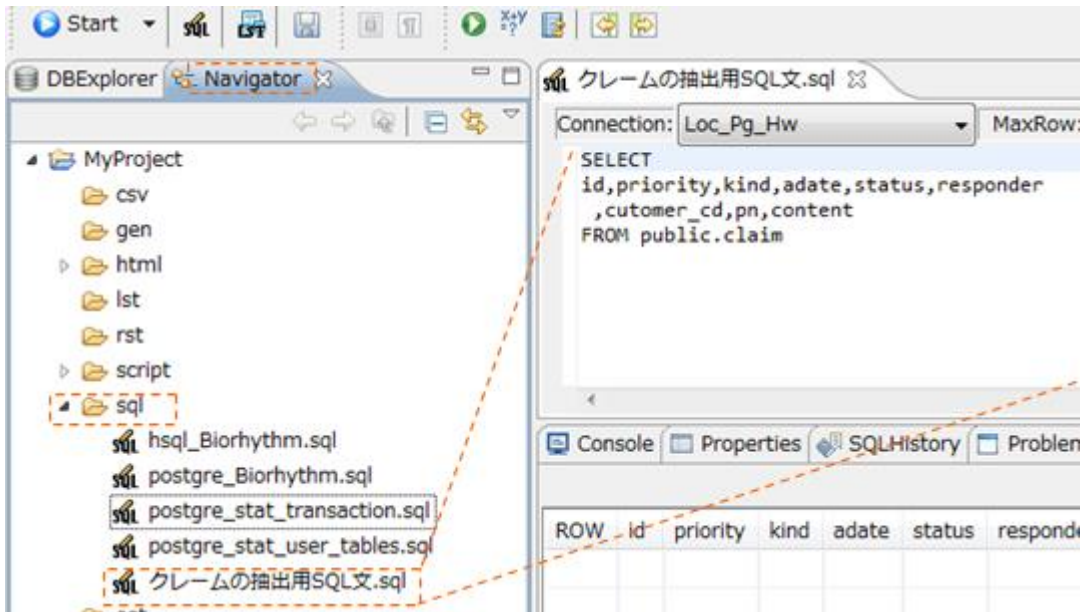
①テーブル名をダブルクリックし、選択子画面を表示します。

条件を指定し、②選択を実行します。

③対象テーブルのデータを選択するためのSQLが設定されたSQLエディタが開くと共に、DBへの選択処理が実施されます。

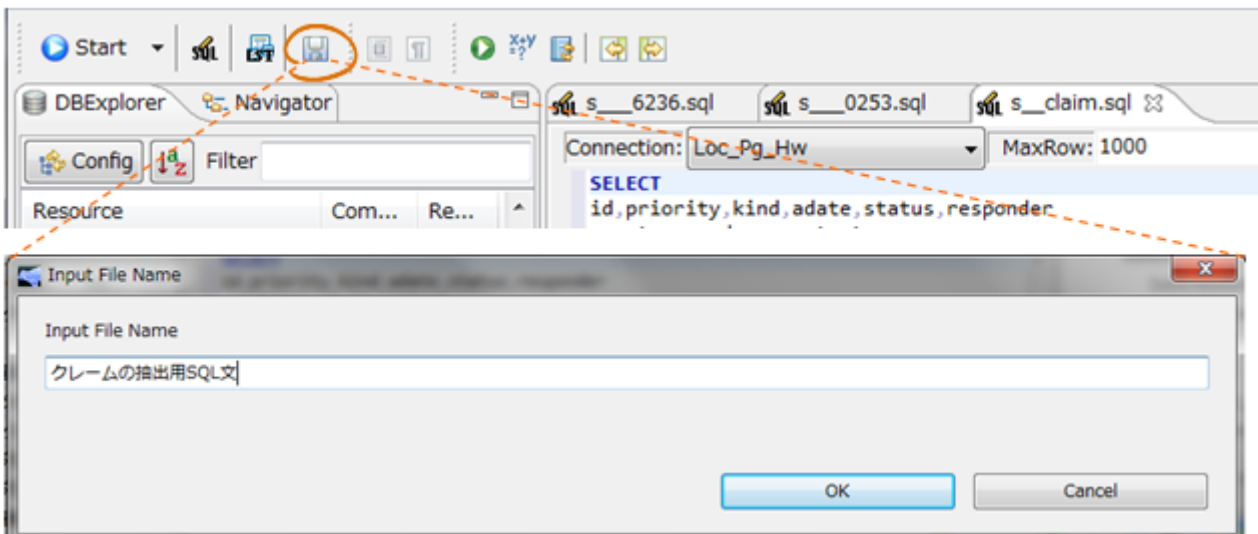


4-1-4 Navigationビューからの表示



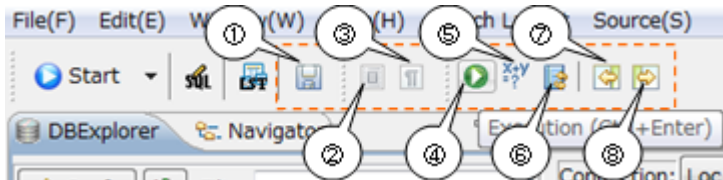
① One Point SQL 文の保存方法

ツールバーの Save アイコンをクリックし、ファイル名を入力します。



4-2 SQL エディタの操作方法

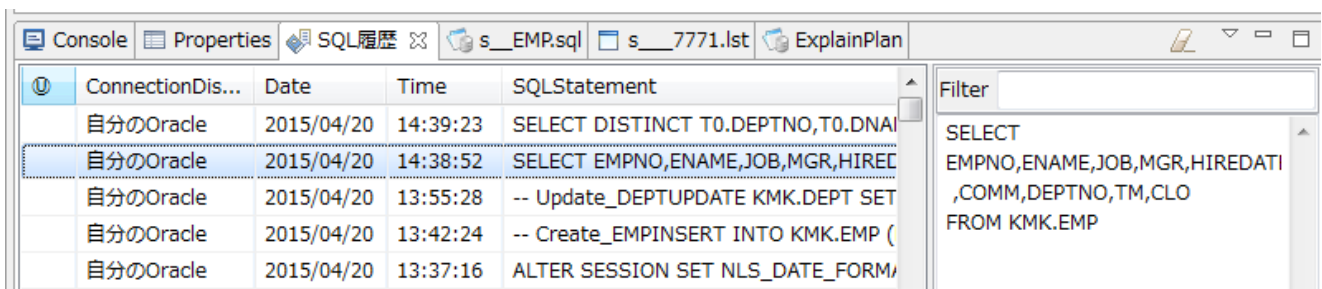
4-2-1 ツールバー



NO	名称	説明	操作方法参照先
1	SQL 文保存	現在エディタに表示中の SQL 文を保存します	
2	ブロック選択	SQL 文を矩形選択します	
3	空白文字可視化	半角、全角スペース、タブを可視化します	
4	SQL 実行	現在エディタに表示中の SQL 文を実行します	4-3
5	実行計画の取得	現在エディタに表示中の SQL 文の実行計画を取得します	4-4
6	SQL 文整形	現在エディタに表示中の SQL 文をフォーマットします	4-5
7	前 SQL 文表示	一つ前に発行に成功した SQL 文を表示します	
8	後 SQL 文表示	一つ後に発行に成功した SQL 文を表示します	

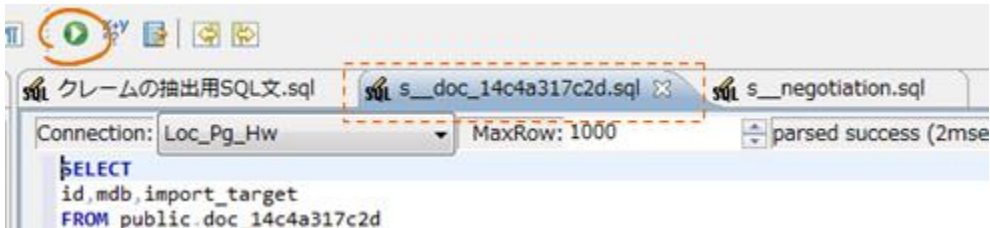
4-2-2 SQL 履歴ビュー

SQL 文の発行に成功した SQL 文を履歴表示します。SQL 文は右側の Filter 部分で部分一致検索出来ます。履歴ビューにて該当 SQL 文をダブルクリックすることで SQL エディタにその SQL 文が転記されます。



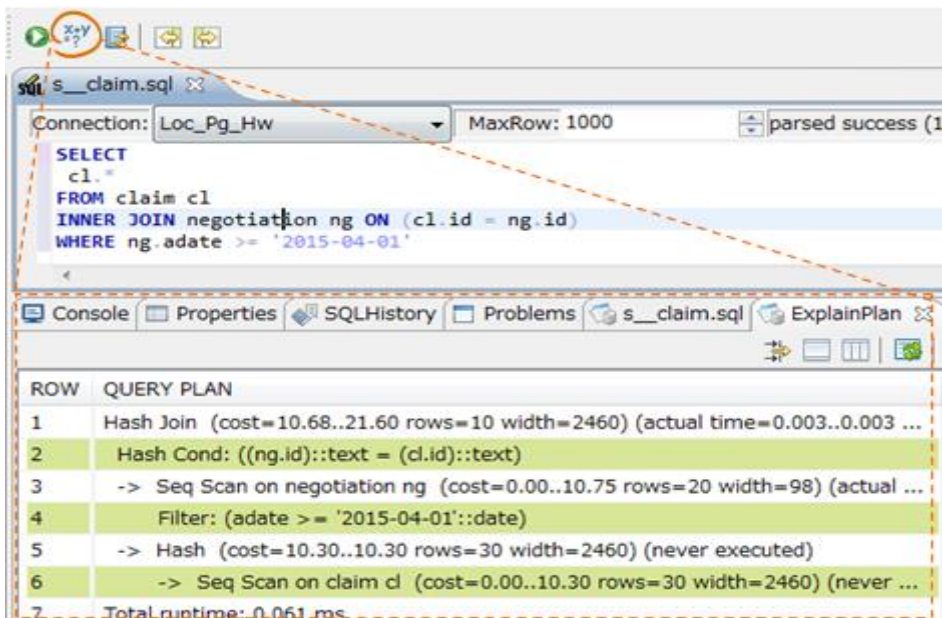
4-3 SQL エディタによる SQL 文の実行

SQL エディタ表示中(アクティブになっている SQL エディタはタブの背景が青くなります。)の SQL 文を実行するにはツールバーの『SQL 実行』ボタンを押下するか『CTRL+ENTER』を押下します。



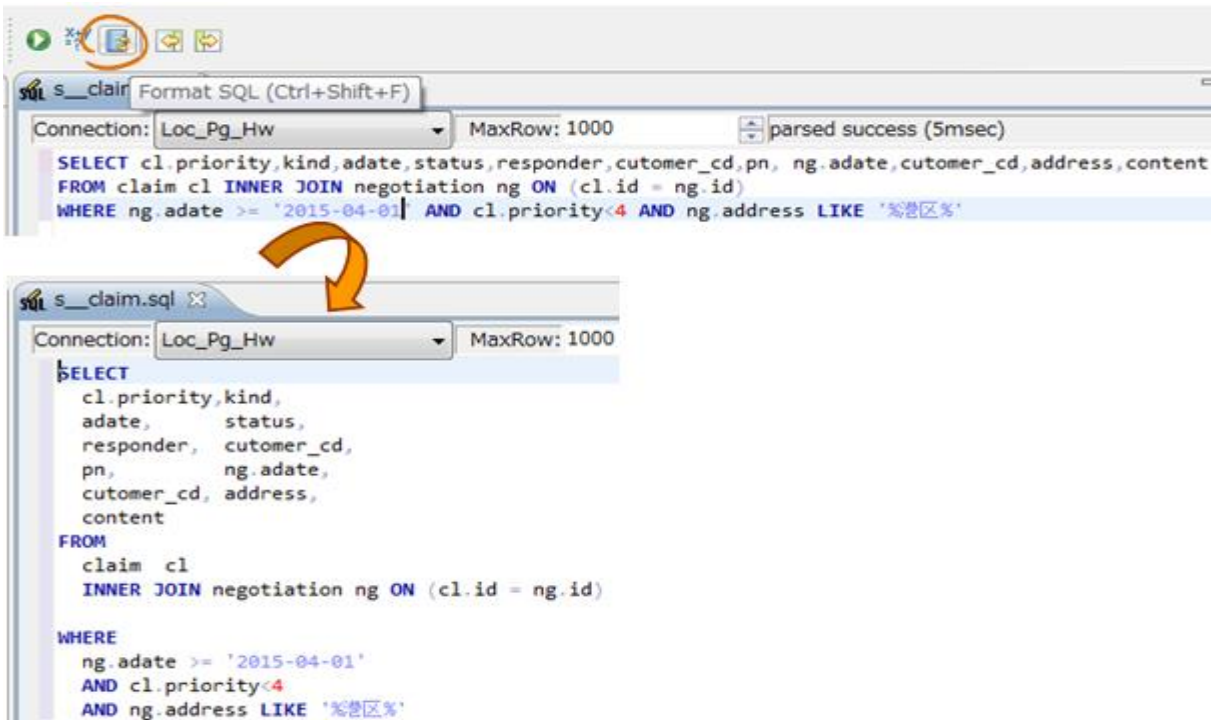
4-4 SQL エディタによる実行計画の取得

SQL エディタ表示中(アクティブになっている SQL エディタはタブの背景が青くなります。)の SQL 文に関し、実行計画を取得するにはツールバーの『実行計画の取得』ボタンを押下します。



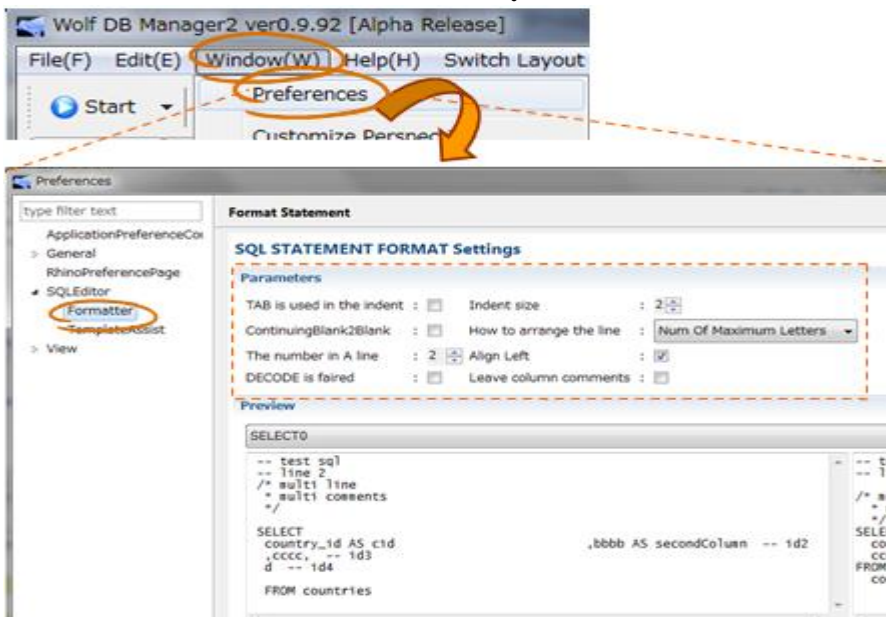
4-5 SQL 文の整形(フォーマット)

4-5-1 整形実行



4-5-2 整形設定

『Window』 - 『Preferences』 --> 『SQLEditor』 - 『Formatter』



5 LST エディタと SQL 文の実行

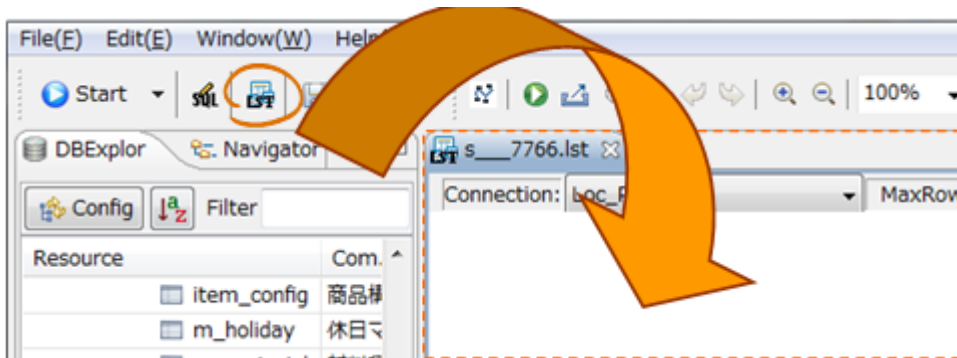
LST は、tableLIST(テーブルリスト)の略であり、複数のテーブルに対し SELECT 文を一括で発行する際に当エディタを利用します。

5-1 LST エディタの表示方法

LST エディタは 3 つの方法で表示することが可能です。

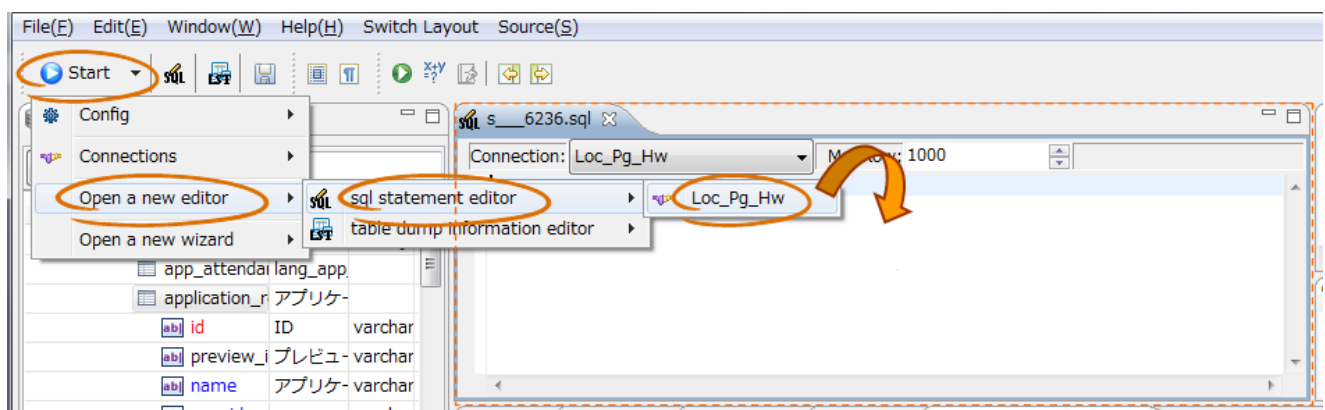
5-1-1 LST アイコンからの LST エディタの新規表示

ツールバーの『LST』アイコンをクリックする

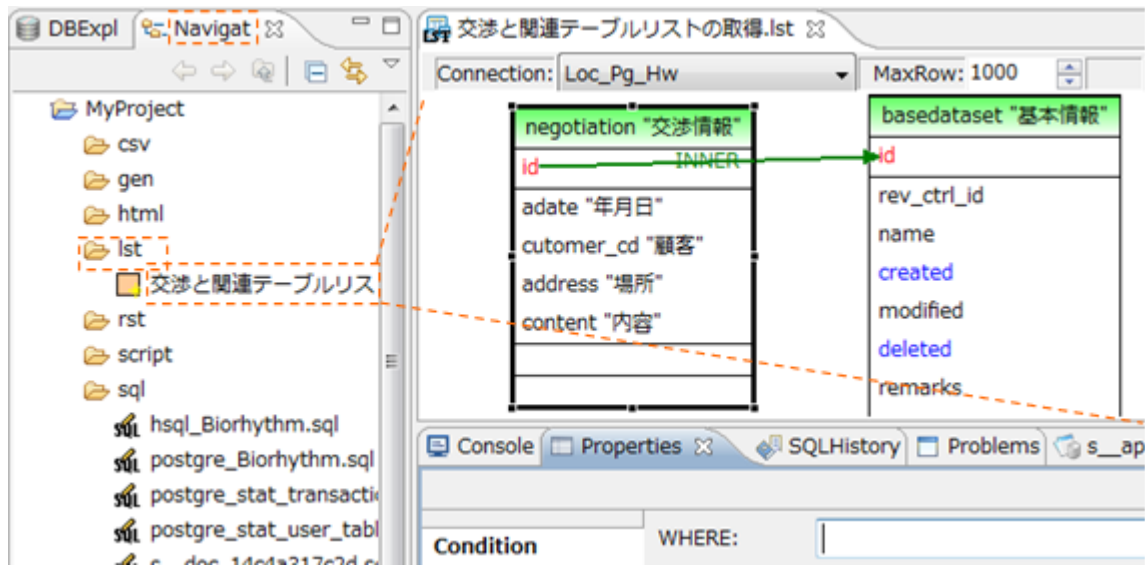


5-1-2 Start ボタンからの表示

ツールバーの『Start』 - 『Open a new editor』 - 『table dump information editor』 - 『接続済コネクション名』をクリックする。

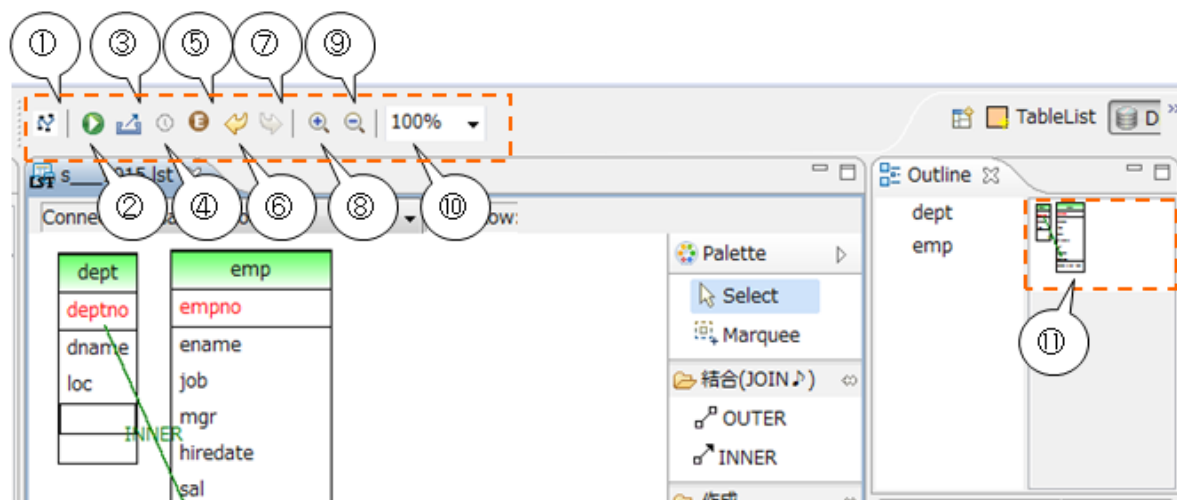


5-1-3 Navigationビューからの表示



5-2 LST エディタの操作方法

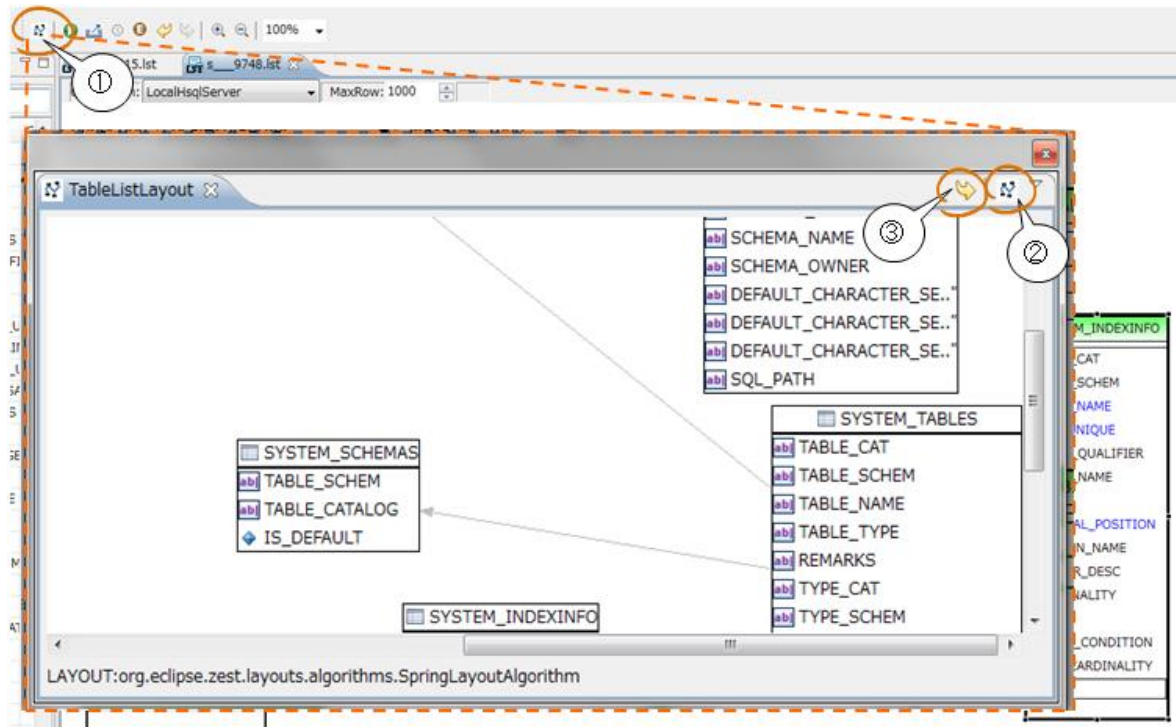
5-2-1 ツールバーとアウトライン



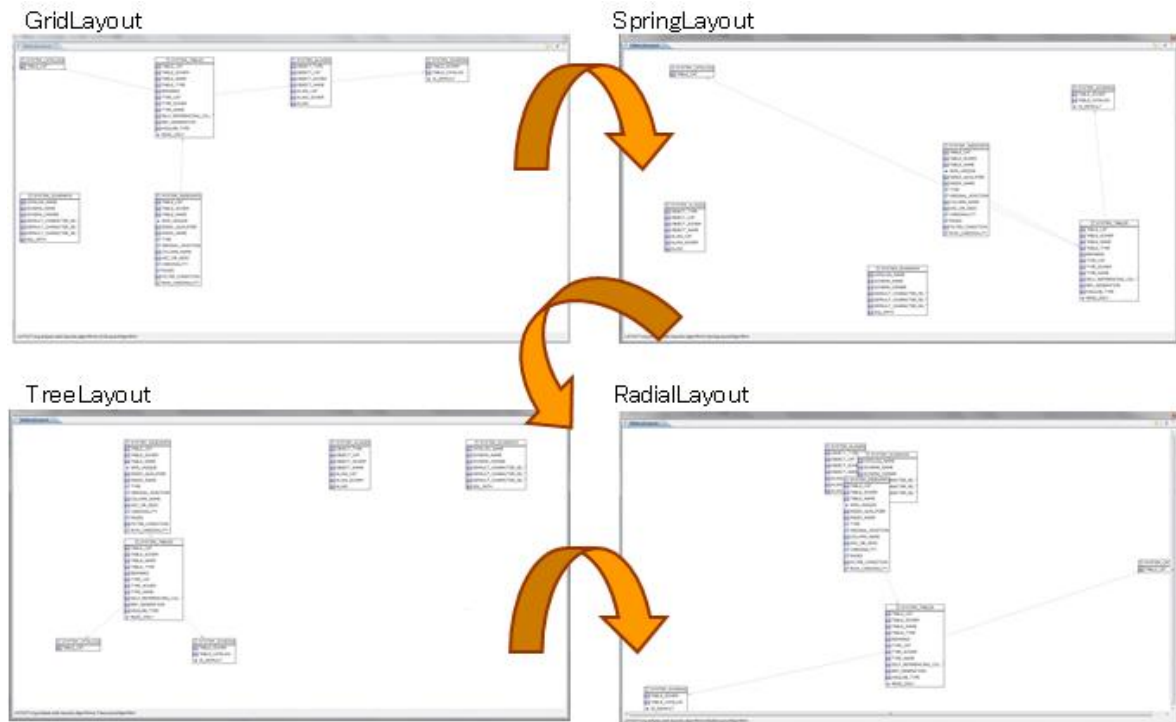
NO	名称	説明	操作方法参照先
1	レイアウト調整	乱雑に配置したテーブル情報を見易く再レイアウトします	5-2-2
2	SQL 実行	設定した検索条件、表示順で SELECT 文を実行します	5-3
3	CSV 出力	SELECT 結果を CSV 出力します	
4	FlashBack 問合せ	Oracle にてフラッシュバック問合せを実行します	
5	ER 図作成	配置したテーブル情報で ER 図を作成します	
6	Undo(戻す)	直前の処理を戻します	
7	Redo(やり直す)	直前の処理をやり直します	
8	拡大	拡大表示します	
9	縮小	縮小表示します	
10	倍率選択	表示倍率を変更します	
11	サムネイル表示	テーブル情報全体が見渡せるサイズで縮小表示します	

5-2-2 レイアウト調整

①レイアウト調整ボタンを押下し、レイアウトの自動調整用ポップアップを表示します。

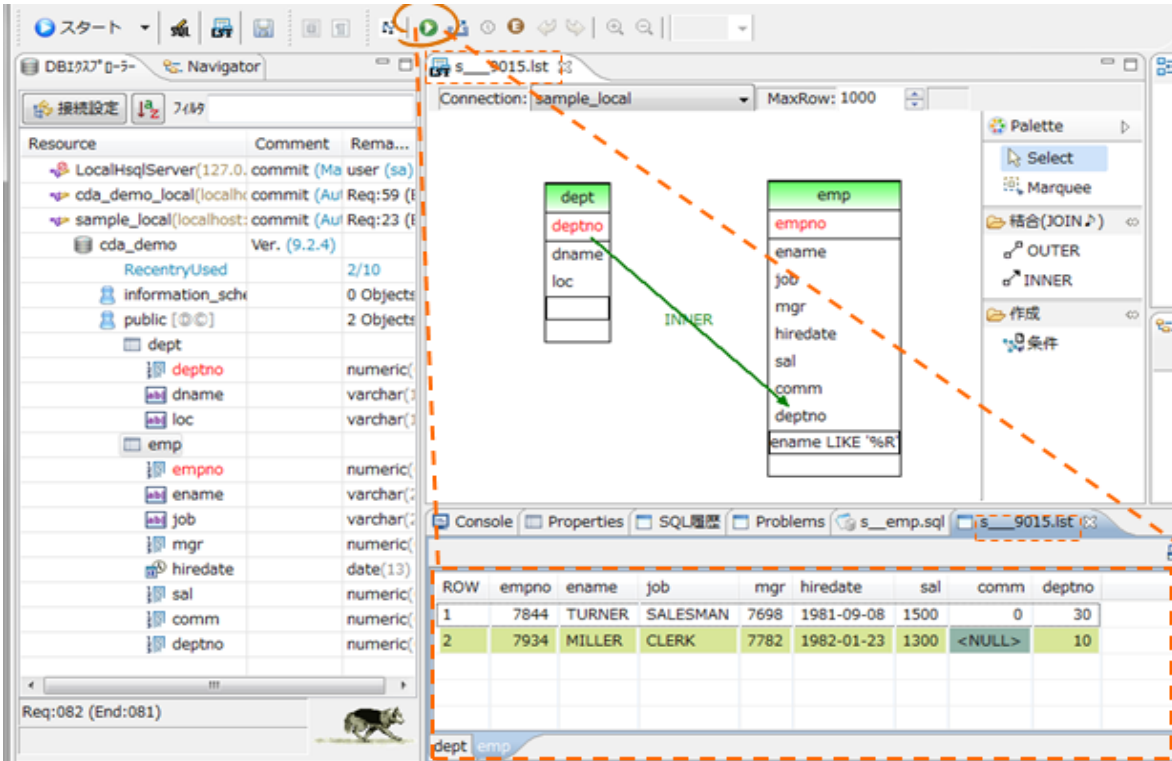


②をクリックする度に下図のように再レイアウトされます。レイアウトを親画面に反映する場合は③を押下します。



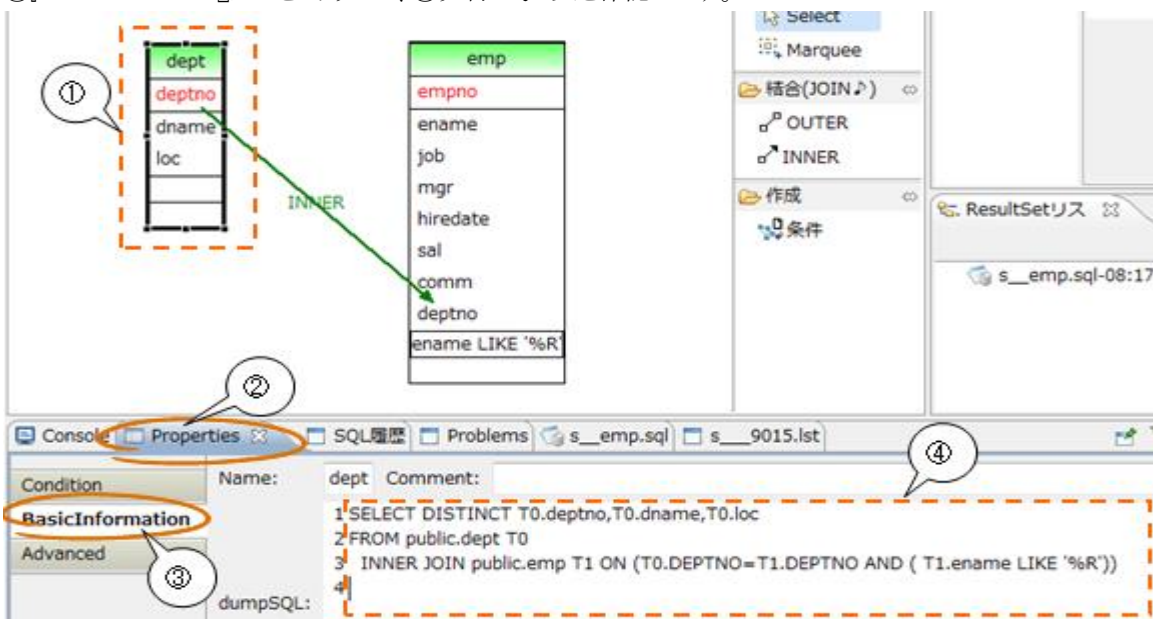
5-3 LST エディタによる SQL 文の実行

LST エディタ表示中(アクティブになっている LST エディタはタブの背景が青くなります。)の SQL 文を実行するにはツールバーの『SQL 実行』ボタンを押下します。



5-3-1 発行する SQL 文の確認方法

- ①対象テーブルを選択し、②『プロパティ』タブを表示します。
- ③『BasicInformation』タブをクリックし、④発行 SQL 文を確認します。



5-3-2 データ抽出条件、表示順、重複除去の指定方法

- ①条件を指定したいテーブルをクリックし、②プロパティタブを表示します。
抽出条件は③に SQL 文の WHERE 句として記述します。
表示順は④に SQL 文の ORDER BY 句として記述します。
重複を除去して抽出する場合は⑤にチェックを入れます。

The screenshot shows the Oracle SQL Developer interface. At the top, there are tabs for 's_DEPT.sql', 's_EMP.sql', 's_4096.lst', and 's_2180.sql'. Below the tabs, the 'Connection' is set to '自分のOracle' and 'MaxRow' is set to '1000'. The main area displays two tables: 'EMP' and 'DEPT'. The 'EMP' table is selected, and its columns are listed: EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO, TM, and CLO. Below the table list, the query text is shown: 'ename LIKE '%R' AND MGR >= 200' and 'SAL DESC, TM ASC'. The 'Properties' tab is active, showing the 'Condition' section with 'WHERE: ename LIKE '%R' AND MGR >= 200', 'ORDER BY: SAL DESC, TM ASC', and 'AS COM' with a checkmark. The 'DISTINCT' checkbox is also checked. Circled numbers 1 through 5 indicate the steps described in the text: 1. Clicking the 'EMP' table, 2. Clicking the 'Properties' tab, 3. Entering the WHERE clause, 4. Entering the ORDER BY clause, and 5. Checking the 'DISTINCT' checkbox.

EMP	DEPT
EMPNO	DEPTNO
ENAME	DNAME
JOB	LOC
MGR	
HIREDATE	
SAL	
COMM	
DEPTNO	
TM	
CLO	

Condition

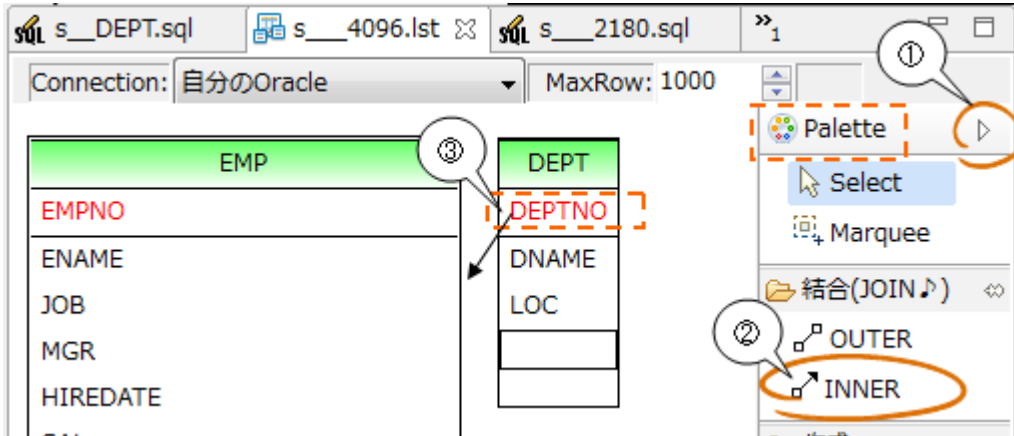
WHERE: ename LIKE '%R' AND MGR >= 200

ORDER BY: SAL DESC, TM ASC

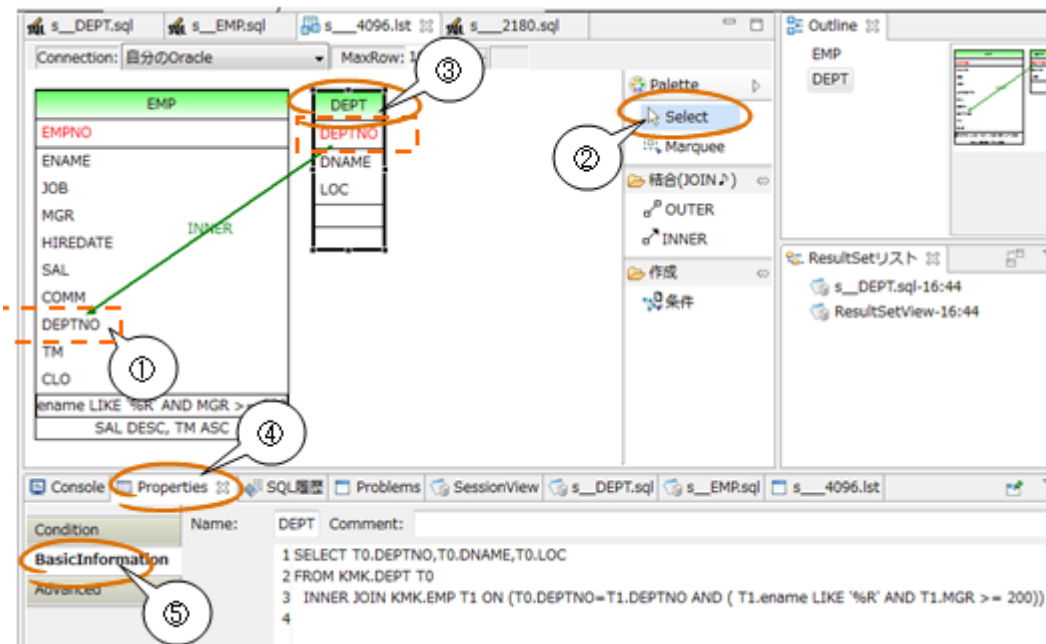
AS COM DISTINCT:

5-3-3 抽出結果に依存したデータ抽出

『Palette』が非表示の場合は①『△』を押下して表示します。DEPT 表で抽出するデータが EMP 表で抽出した結果の DEPTNO に存在するものだけを抽出したい場合は②結合 INNER を選択し、③DEPT 表の DEPTNO 列をクリックします



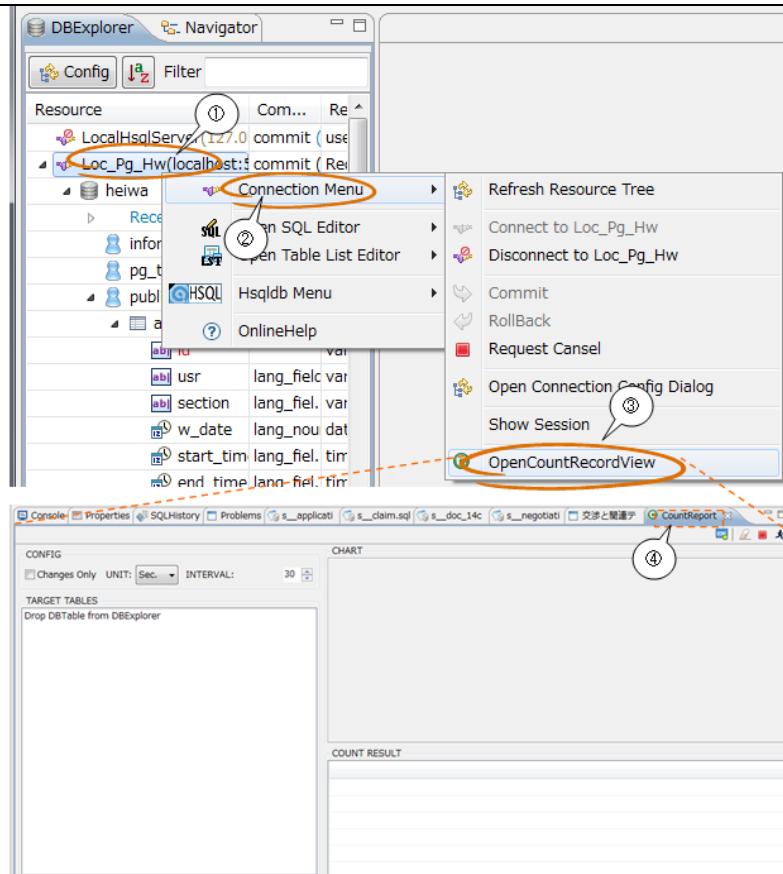
DEPT 表の DEPTNO 列から結合線を伸ばし①EMP 表の DEPTNO 列を選択し、結合させます。結合の指定はもう必要ないので②『Select』モードに変更します。③DEPT 表の抽出条件を確認する為、④『Properties』タブを表示し、⑤『BasicInformation』から SQL 文を確認します。



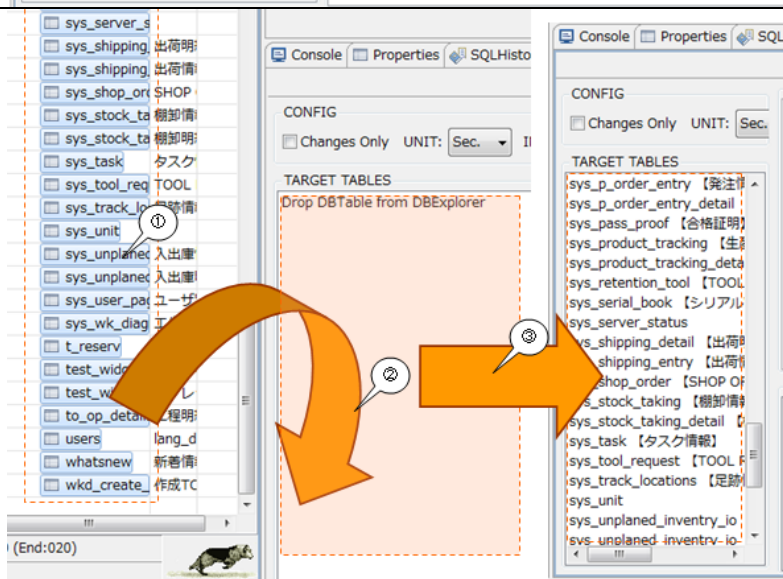
6 逆引き

6-1 任意のテーブルを対象にレコード数の増減を確認したい

未知のシステムに於いて CRUD の設計書も無い場合に、画面操作の前後で変更が発生したテーブルを簡単に確認する方法を示します。

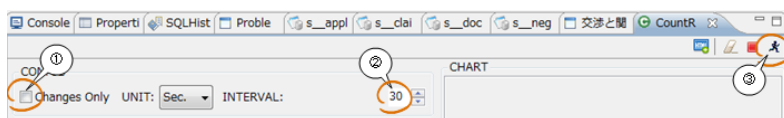


- 1 ①接続定義を右クリックします
②接続メニューを選択します
③カウントレポートビューを選択します
④ビューが開いたことを確認します



- 2 増減監視対象テーブルの指定
①対象となるテーブルを選択し、②『ターゲットテーブル』へ Drag & Drop する
③リストに登録された事を確認する

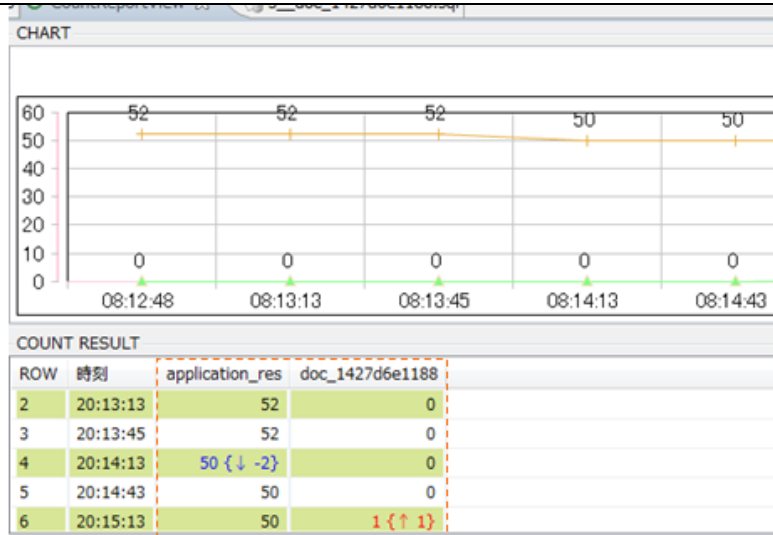
(SHIFT キーや CTRL キーを押下しつつ
選択する事で対象テーブルを複数指定で
きます)



- 3
- ①変更のあったテーブルに絞って表示する場合にチェック
 - ②監視間隔を指定
 - ③ 監視実行

監視対象テーブルに対し、データの増減を伴う操作を実施

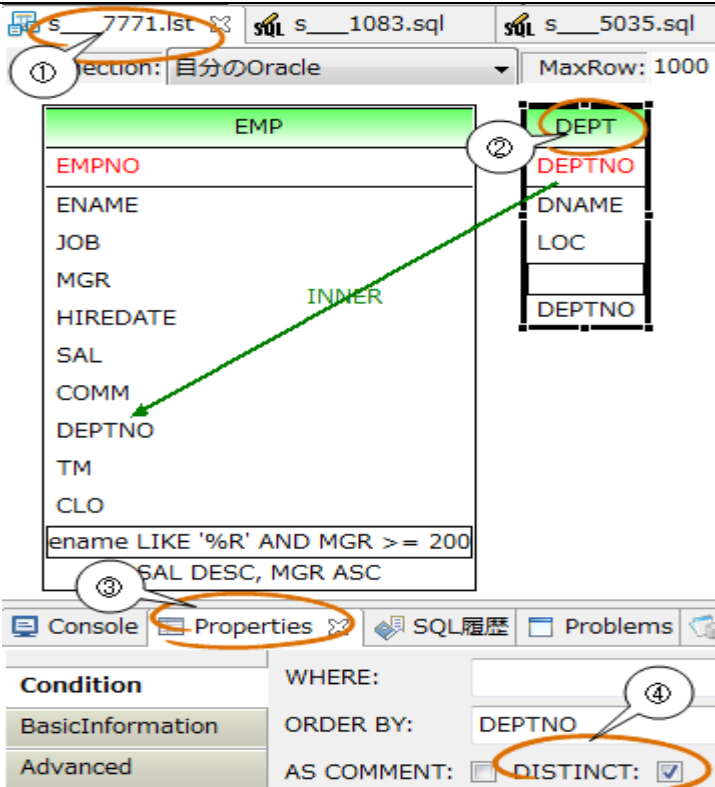
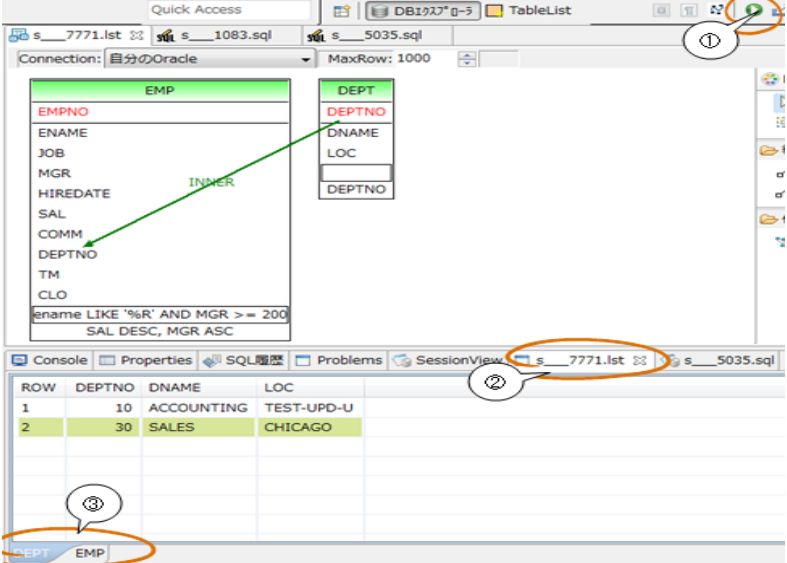
4



- 5
- 指定間隔でレコード数の増減のあったテーブルだけが増減数、総レコード数と共に表示されます

6-2 データの変更内容を確認したい、エビデンスとして残したい

CRUD マトリックスに従い、データの Create (新規追加)、Update (更新)、Delete (削除) の Evidence(証跡)を確認、作成する方法を示します。

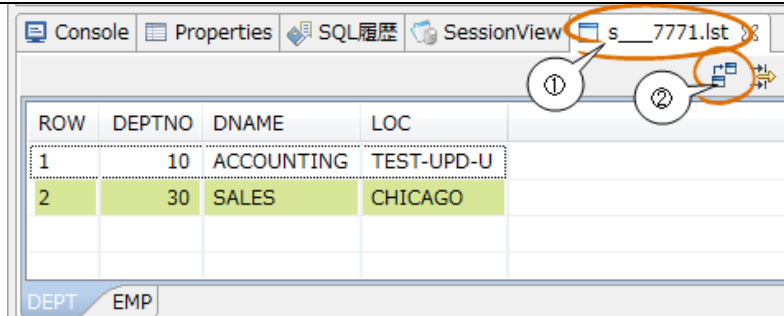
<p>『5-1 LST エディタの表示方法』に沿ってLST エディタを表示</p>	<p>1 LST エディタを表示します</p>
<p>『5-3 LST エディタによるSQL文の実行』に沿ってエビデンス抽出用の条件、表示順を指定</p> 	<p>2 条件、表示順の指定</p> <p>3 データを見易い形で取得する為、重複を省くように LST エディタに条件を指定します。(参考:5-3-2-⑤)</p> <p>①データ抽出用のLST エディタをアクティブにします ②DEPT 表を選択します ③Properties を表示します ④Distinct を指定します</p>
	<p>4 データの変更が行われる前の状態を Wolf に取得します。</p> <p>①SQL の実行 ②結果の選択 ③各表毎に表示される抽出結果の確認</p>

```

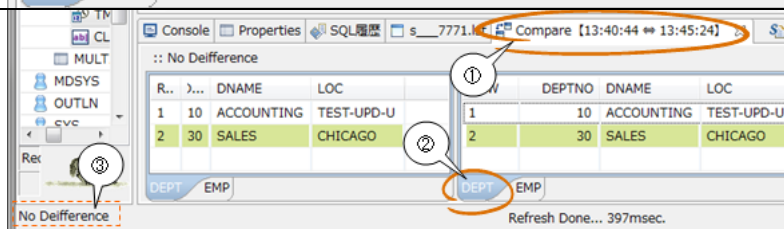
-- Create_EMP
INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO, TM)
VALUES (9441, 'KINSR', 'PM', 400, '1970-01-01', 200, NULL, 10, sysdate);
INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO, TM)
VALUES (9442, 'ASAHR', 'PL', 350, '1972-01-01', 220, NULL, 30, sysdate);
-- Update_EMP
UPDATE EMP SET SAL=200 WHERE EMPNO=7844;
-- Delete_EMP
DELETE FROM EMP WHERE EMPNO = 7934;

```

5 左記の通り、EMP 表への変更が伴う操作が実行された後



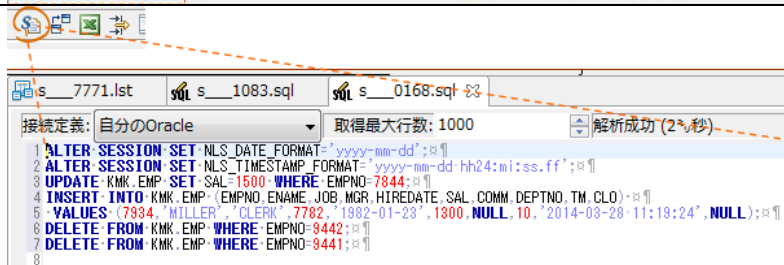
6 ①LST 取得結果の②同一 SQL での比較ボタンを押下します



7 ②『DEPT』表は未変更だったことを確認します

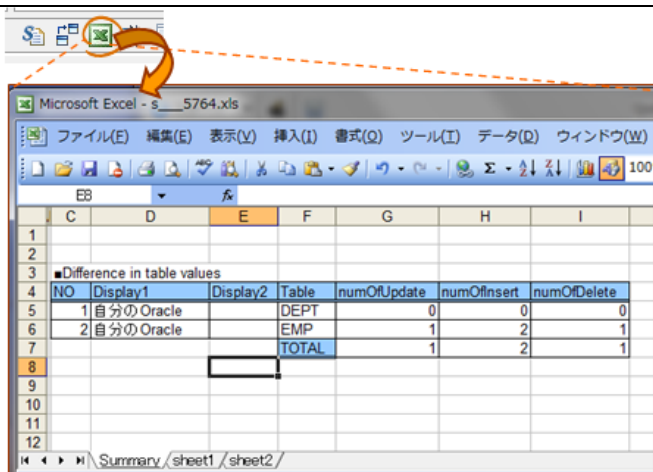


8 ①『EMP』表には②UPDATE が 1 件、INSERT が 2 件、Delete が 1 件あることを確認します



9

① One Point
Generate Restore Button を押下するとテーブルの状態を元に戻す SQL が生成されます。

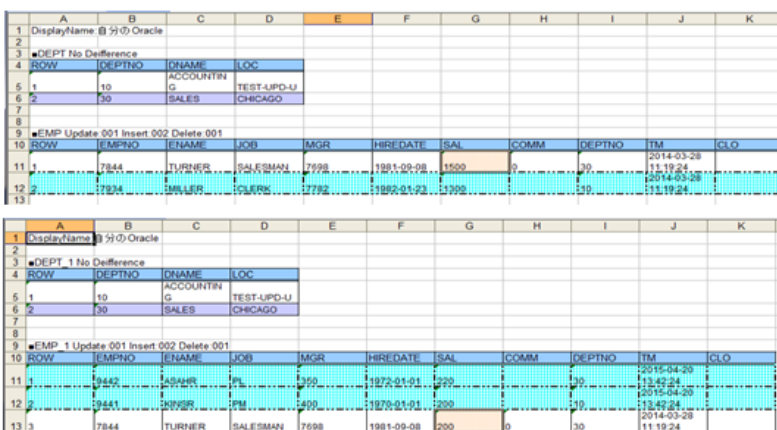


10 Excel 出力 (エビデンス生成)

Summary と変更前後のシートが生成されます。

UPDATE => 赤いセル

INSERT or DELETE => 水色セル



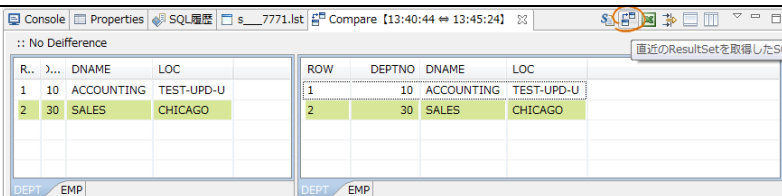
— Update DEPT

UPDATE DEPT SET LOC='ESPRESSO' WHERE DEPTNO=10;

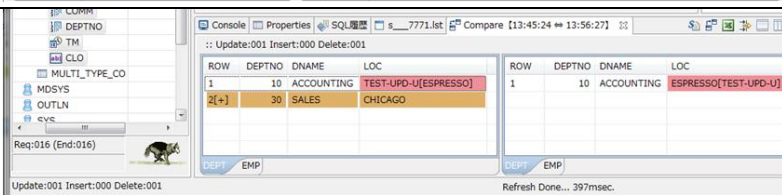
— Delete DEPT

DELETE FROM DEPT WHERE DEPTNO = 30;

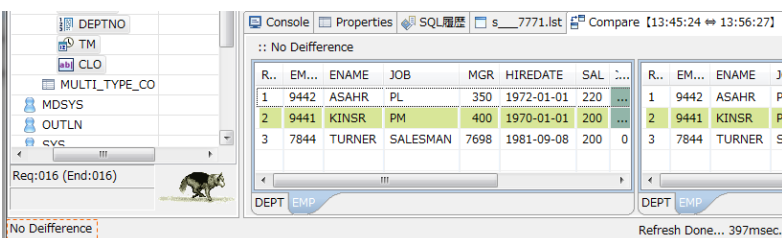
11 左記の通り、DEPT 表への変更が伴う操作が実行された後

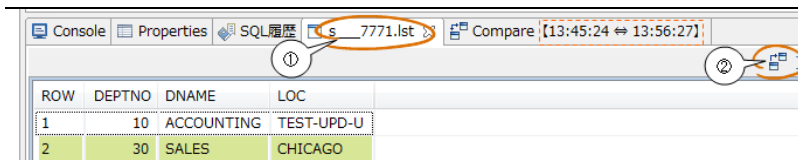


12 #6 で作成したビューにて、更に『比較』ボタンを押下します



13 『DEPT』表にのみ変更がなされたことを確認します

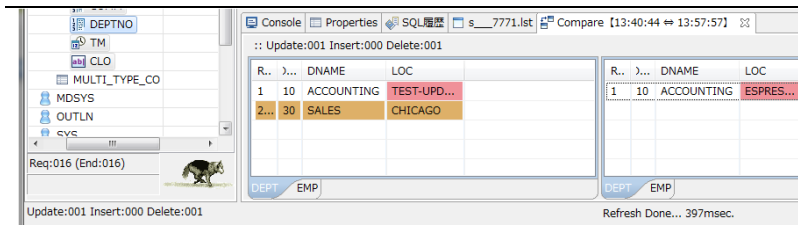




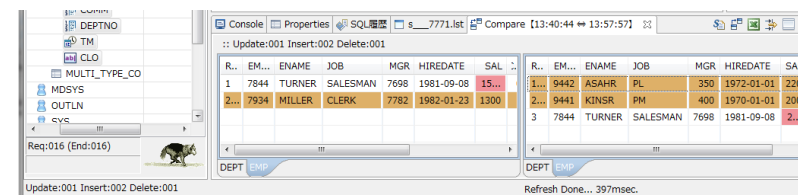
14 2テーブル分の変更に対する差分の取得

①#12で入替た比較ビューではなく、#4で作成したビューを選択します(変更前の最初のデータ)

②『比較』ボタンを押下します



15 2テーブルに対する差分が取得されたことを確認します



C	D	E	F	G	H	I
1						
2						
3	■Difference in table values					
4	NO	Display1	Display2	Table	numOfUpdate	numOfInsert
5	1	自分のOracle		DEPT	1	0
6	2	自分のOracle		EMP	1	2
7				TOTAL	2	2
8						

16 『Excel』出力ボタンを押下し、エビデンスとして保存します

A	B	C	D	E	F	G	H	I	J	K
1	DisplayName:自分のOracle									
2										
3	■DEPT Update:001 Insert:000 Delete:001									
4	ROW	DEPTNO	DNAME	LOC						
5	1	10	ACCOUNTING	TEST-UPD-U						
6	2	30	SALES	CHICAGO						
7										
8	■EMP Update:001 Insert:002 Delete:001									
9	ROW	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	ITM
10	1	7844	TURNER	SALESMAN	7698	1981-09-08	1500	0	30	2014-03-28 11:19:24
11	2	7934	MILLER	CLERK	7782	1982-01-23	1300	0	30	2014-03-28 11:19:24
12	3	7844	TURNER	SALESMAN	7698	1981-09-08	200	0	30	2014-03-28 11:19:24
13										
14										
15										
16										
17										
18										
19										
20										
21										
22										
23										
24										
25										
26										
27										
28										
29										
30										
31										
32										
33										
34										
35										
36										
37										
38										
39										
40										
41										
42										
43										
44										
45										
46										
47										
48										
49										
50										
51										
52										
53										
54										
55										
56										
57										
58										
59										
60										
61										
62										
63										
64										
65										
66										
67										
68										
69										
70										
71										
72										
73										
74										
75										
76										
77										
78										
79										
80										
81										
82										
83										
84										
85										
86										
87										
88										
89										
90										
91										
92										
93										
94										
95										
96										
97										
98										
99										
100										

6-3 データを別のデータベースへ移行したい

別のデータベースに任意のスキーマ、テーブルを保持しているレコードも含めて複製します。データを複製するのが主目的であり、インデックスや制約は複製されません。

参考頁

<http://yonsama-blog.blogspot.jp/2011/10/wolfdatabase2-rcp-how-to-use-5.html>

6-4 新しいバージョンの Wolf に古いバージョンの設定を引継ぎたい

当操作により『2-2 DB 接続初期設定』にて設定する情報を引継ぎます。
尚、Ver0.x.x から Ver1.x.x へは引継ぎません。

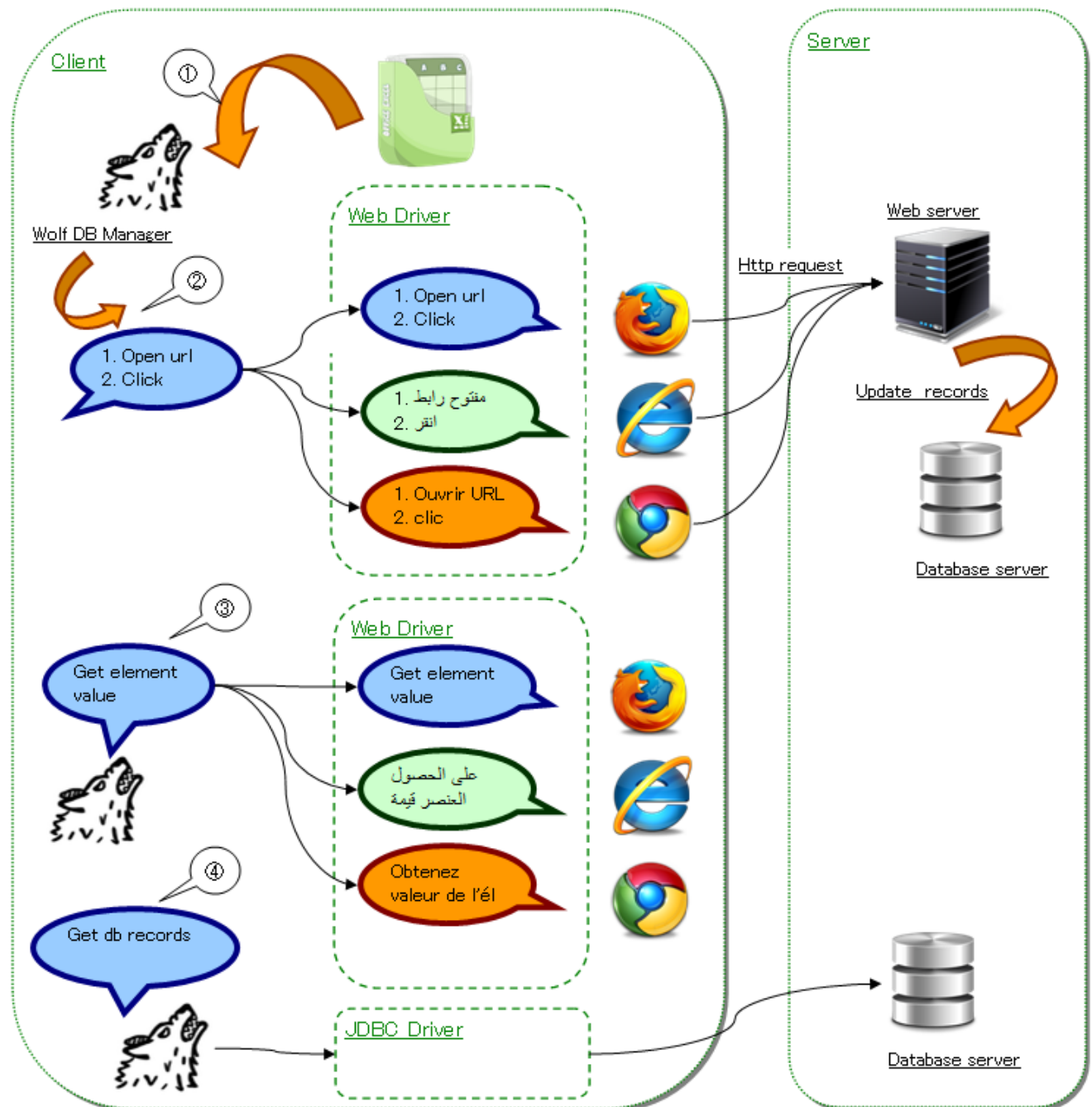
	<p>1 メニューバーの①『ウィンドウ』から②『設定(Preferences)』を押下します</p>
	<p>2 開いた子画面にて①『旧Wolfからの移行』を選択します ②『旧バージョン(Old Ver...)』のファイル選択ボタンを押下します</p>
	<p>3 ①旧バージョンの「実行ファイル(exe)や設定フォルダ(w_config)」を含むフォルダを指定します ②o.k.を押下します</p>
	<p>4 o.k.を押下します</p>
	<p>5 o.k.を押下すると定義が引継がれたアプリケーションが再起動します</p>

7 テストシナリオランナー

テストシナリオランナーは Excel に記述したシナリオに沿って WEB ブラウザを自動操作します。また、自動操作によって生じた DB テーブル、ブラウザ上の項目の値の変化を任意に検証することも可能です。

7-1 概要

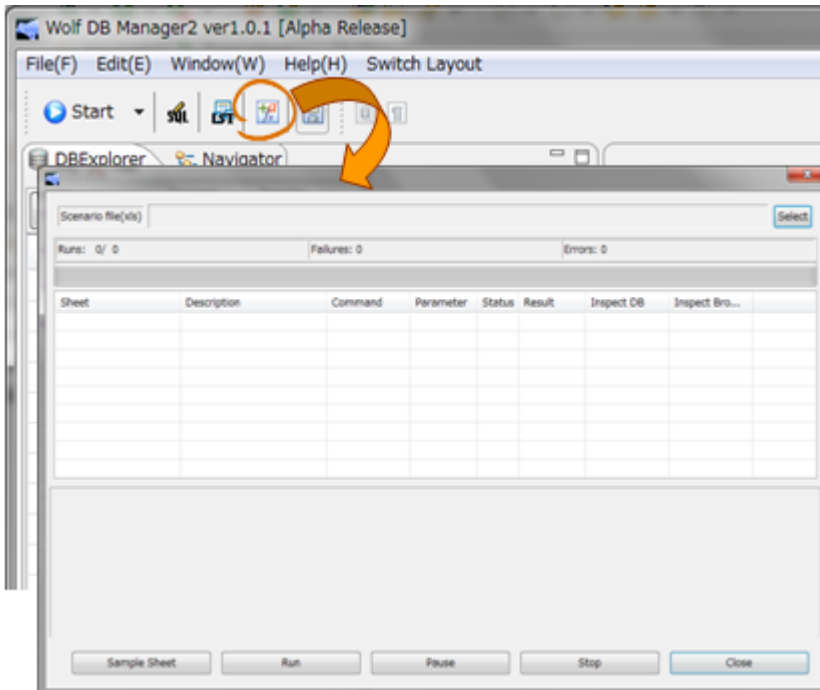
- ① Excel ファイルに記述したブラウザの操作命令に従い、② ブラウザ毎の WebDriver を介してブラウザを自動操作します。尚、Firefox ブラウザだけは WebDriver ファイルを別途用意する必要はありません。
- ③④、②の操作により生じたブラウザ上の値、DB テーブルの値を取得し、期待値との比較検証を行います。



7-2 操作方法

7-2-1 起動方法

ツールバーの『Scenario Runner』アイコンをクリックする



7-3 シナリオ実行初期設定

7-3-1 設定

NO	プロパティ	内容	設定例	備考
1.1.1	base url	相対パス指定時の基準 URL	http://localhost:8080/hoge	
1.2.1	exit on failure	エラー発生時に停止する	true	
2.1.1	output path	エビデンス保存フォルダ	C:\tmp\wolf_evidence	指定することをお勧めします
2.2.1	take pictures	スクリーンショット保存タイミング	auto	auto / manual
2.2.2	take_on_changed_screen	画面遷移時に自動取得	true	2.2.1 が auto の場合に有効
2.2.3	take_on_clicked	クリック時に自動取得	true	2.2.1 が auto の場合に有効
2.2.4	take_on_sended_keys	キー入力時に自動取得	true	2.2.1 が auto の場合に有効

7-4 シナリオの自動操作に関する記述

7-4-1 (主にブラウザへの)命令

NO	アクション	説明	パラメータ1	同2	同3
1	openUrl	ページを開く	対象サイト URL		
2	back	戻る			
3	forward	進む			
4	refresh	再表示			
5	clear	クリア			
6	click	クリック			
7	doubleClick	ダブルクリック			
8	moveToElement	マウスホバー			
9	contextClick	右クリック			
10	selectByIndex	インデックスで選択	0 から始まる option 要素のインデックス	select 要素に multiple 属性がある場合、2 つ目、3 つ目の選択値を指定する	
11	selectByVisibleText	テキスト値で選択	プルダウンの表示内容 内容		
12	selectByValue	値で選択	option 要素の value 値		
13	getAttribute	属性の取得			
14	getCssValue	CSS 値の取得			
15	getText	Text の取得			
16	submit	サブミット			
17	sendKeys	キー入力	入力内容		
18	takesScreenshot	画面キャプチャ	画像ファイルパス (未指定の場合はエビデンスフォルダに保存)		
19	takesAllScreenshot	デスクトップ全画面キャプチャ	同上		
20	waitForInvisible	要素が非表示になるまで待機	タイムアウト秒数 (未指定の場合は 60 秒)		
21	switchTo	対象画面を切替る			
22	switchToDefaultContent	TOP 画面に操作を戻す			
23	switchToConfirmAndAccept	確認ダイアログの OK 押下			
24	switchToConfirmAndDismiss	確認ダイアログの NG 押下			
25	switchToErrorAndAccept	エラーダイアログの OK 押下			
26	setWindowSize	ウィンドウサイズを変更	width,height 又は vga,xga,sxga など		
27	inspectBrowser	ブラウザの内容を検証する			
28	inspectDB	DB の内容を検証する			
29	gotoNoWhenExistsSelector	セレクタの要素が存在する場合に(指定シートの)指定 NO までスキップ	スキップ先 NO	スキップ先シート名 (未指定の場合は自シート)	
30	~ExistsMultipleSelectors	~複数存在する場合にスキップ			
31	~NotExistsSelector	~存在しない場合にスキップ			
32	setFile	ファイル指定	ファイルの絶対パス		
33	deleteAllCookies	クッキーの全削除	対象サイト URL		
34	execute_script	JavaScript を実行する	スクリプト		
35	execute_async_script	JavaScript を非同期で実行する	スクリプト		

7-4-2 パラメータ（変数）の展開

パラメータには固定文字列だけでなく、変数も指定可能です。

変数は`{xxx}`という形式で半角文字のドル、波括弧、括弧内に半角英数記号で記述します。

NO	変数名	内容	備考
1	yyyymm	シナリオ実行開始時の年月	今月は <code>{yyyymm}</code> ⇒ 今月は 201505
2	yyyymmdd	同、年月日	today is <code>{yyyymmdd}</code> ! ⇒ today is 20150504!
3	yyyymmddHHMM	同、年月日時分	Now, <code>{yyyymmddHHMM}</code> ⇒ Now, 201001012359
4	yyyymmddHHMMss	同、年月日時分秒	<code>{yyyymmddHHMMss}</code> ⇒ 19981130102030
5	HHMMss	同、時分秒	<code>{HHMMss}</code> ⇒ <code>_201245_</code>
6	HHMM	同、時分	<code>{HHMM}</code> ⇒ 0123
7	user_name	PC ログインユーザ名	Who am i <code>{user_name}</code> . ⇒ Who am i Wolf.
8	today	今日 (yyyy-mm-dd)	<input type='date'> の value に利用します。
9	now	今 (yyyy-mm-dd HH:MM:ss)	<input type='datetime'> の value に利用します。
10	改行	改行文字(ENTER)キー	改行コード「 <code>\n</code> 」として利用します
11	確定		
12	ENTER		

7-5 自動操作に起因するデータベース検証処理の記述（任意）

検証処理の記述は任意です。従って、記述しなくても自動操作に影響はありません。

7-5-1 検証アクション

利用可能な検証アクションは以下の3つです。

NO	検証アクション	内容	備考	参照先
1	count_query	集計関数(COUNT)結果確認		
2	count_table	指定テーブル件数確認		
3	select_query	SELECT 結果の確認		

7-5-2 select_query (SELECT 結果の確認) アクション

ブラウザ操作の前後のタイミングにて別シートに記載した SELECT 文を発行し、その差異を検証します。
検証処理は⑥のパラメータの内容に従い実施し、結果を Excel、Txt にて出力します。

The screenshot shows two sheets. The top sheet, 'Database Inspection', has columns BA through BJ. Row 1 contains labels: action, table, query sheet, bind parameters (with sub-labels 1-5), expect, and parameter. Row 2 contains values: select_query, #cnt_sys_drawing_ctrl, CIF10800, cad, and issue_c_or_u_or_d u > 30, c=0. The bottom sheet, 'Query Sheet', has columns A, B, and C. Row 2 (Description) contains '図面管理情報テーブルを得意先とファイルタイプで縛って抽出する'. Row 3 (Primary key) contains 'num_of_drawing_info, part_info_name'. Row 4 (Query) contains 'SELECT * FROM sys_drawing_ctrl WHERE customer_cd = ? AND file_type = ?'. A red circle highlights the '?' in the WHERE clause, with arrows pointing to the '1' and '2' bind parameters in the 'Database Inspection' sheet.

NO	項目	内容	備考
1	action	検証アクション	「select_query」固定
2	table	テーブル名	未記入固定
3	query_sheet	SELECT 文記述シート名	左記のとおり。(シート名は必ず#ではじめる)
4	bind parameters	上記 SELECT 文のバインド値	左記のとおり。(?の出現順に 1,2,3...と利用してください)
5	expect	期待値の比較演算子	「issue_c_or_u_or_d」固定
6	parameter	期待値	右記書式の組合で指定。(c u d)(> = < < = [0-9]+)? c : create : 行の追加, u : update : 値の変更, d : delete : 行の削除 例) ・『c >= 100, u = 1, d <= 100』 ⇒ 100 行以上の行追加、削除、1 箇所の更新 ・『c, d』 ⇒ 1 行以上の追加、削除があり、値の更新は無し ・『u >= 50』 ⇒ 50 箇所以上の更新があり、削除、追加は無し ・『』 (空文字) ⇒ 操作前後で SELECT 結果に何も変化が無し
7	Description	SELECT 文の説明	任意入力
8	Primary key	SELECT 項目の一意キー	任意入力 (省略時は取得した行番号を一意キーとして値の比較を行います)
9	Query	SELECT 文	バインド値は?で指定する

7-6 自動操作に起因するブラウザ検証処理の記述（任意）

検証処理の記述は任意です。従って、記述しなくても自動操作に影響はありません。

7-6-1 ブラウザの内容変化に伴う検証

NO	プロパティ	内容	設定例	備考
1	dialog_text	ダイアログの表示文言		
2	element_count	要素の数		
3	element_text	要素のテキスト		
4	element_value	要素の値		
5	window_name	ページ名		

8 FAQ

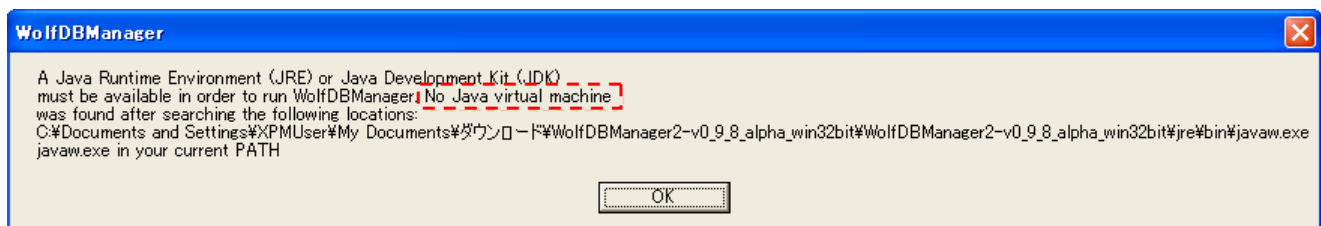
8-1 起動しません

8-1-1 JavaVM が未インストール状態により起動しない

御利用の WindowsPC に JavaVM が未インストール状態の場合、御利用できません。

Oracle のサイト「<https://www.java.com/ja/download/manual.jsp>」より、JavaVM をインストールしてください。

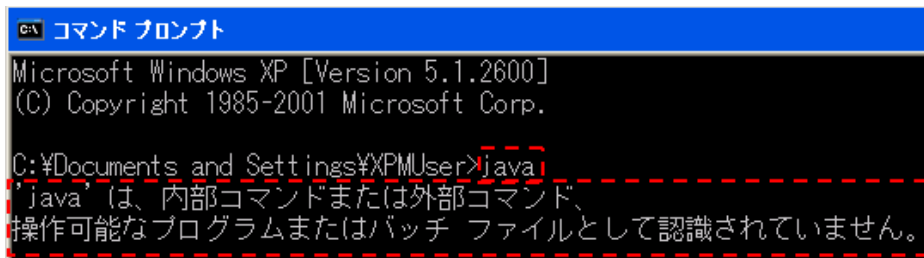
【起動時エラーメッセージ】



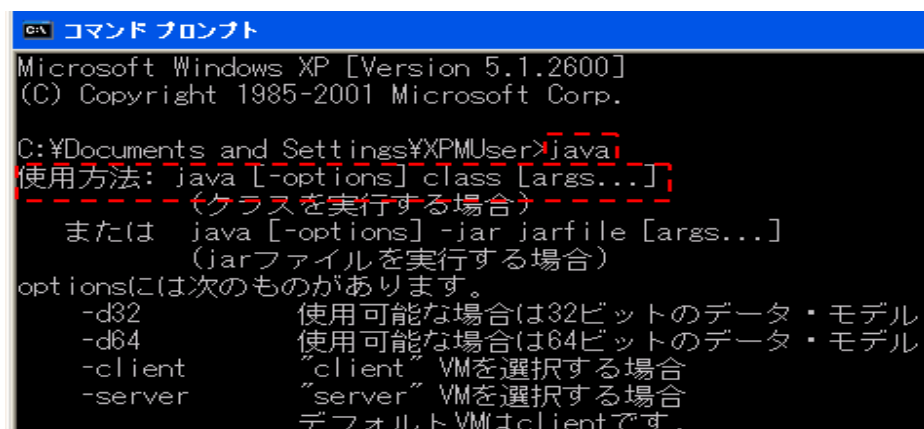
① One Point

コマンドプロンプトでの JavaVM インストール状況確認方法。 java<改行>と打鍵することで確認できます。

<未インストール状態>

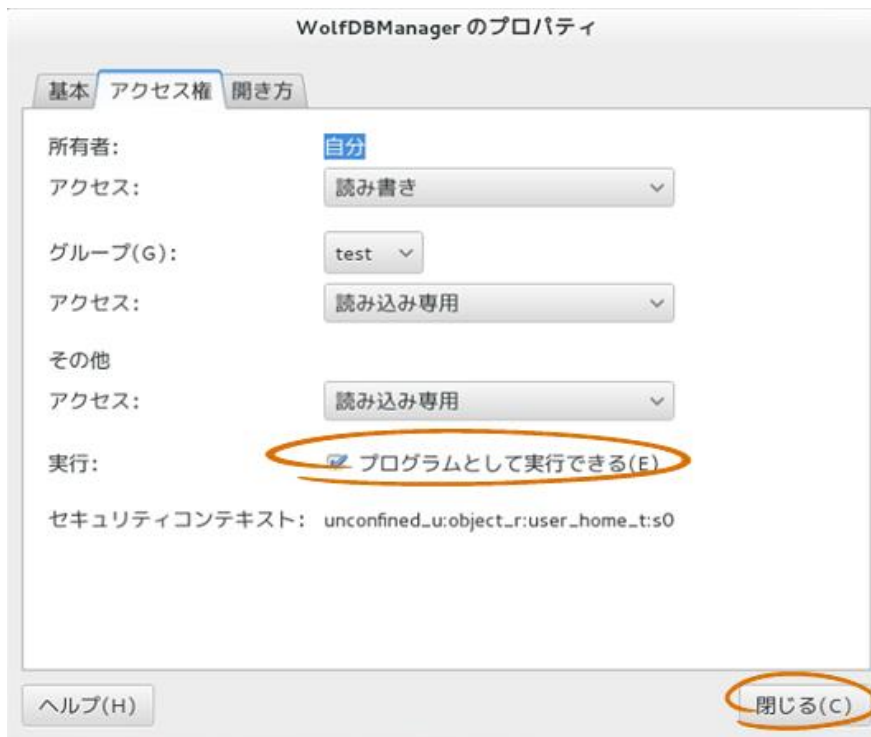
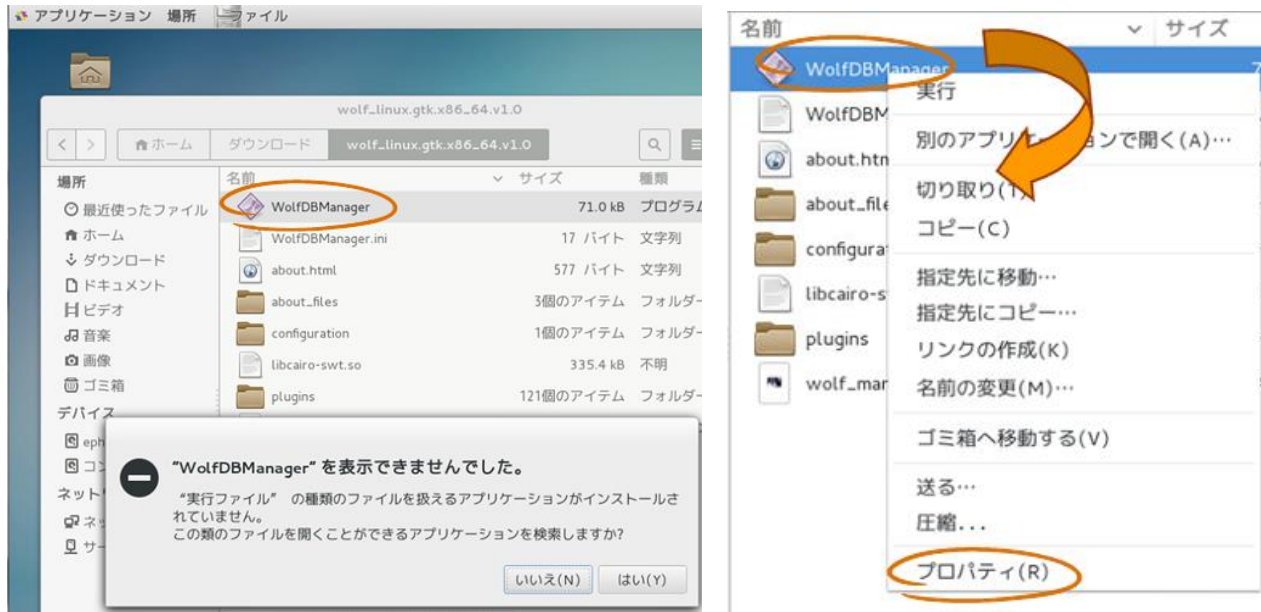


<インストール済状態>



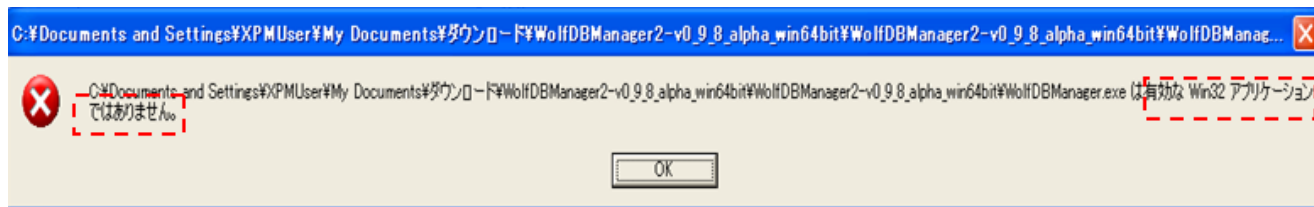
8-1-2 実行権が無いので起動しない

Linux の場合、ダウンロードしたファイルに実行権限を付与する必要があります。



8-1-3 32bitOS の PC にて 64bit 用のアプリケーションを実行することで起動しない

下図のように『有効な Win32 アプリケーションではありません』と表示される場合、ダウンロードしたファイルが間違えています。32bit 版のファイルを御利用ください。



8-1-4 利用方法をまとめたページは他にありませんか？

下記サイトをご利用ください。

http://yonsama-blog.blogspot.jp/search/label/wolf_db_manager

<http://www.sepia.dti.ne.jp/yonsama/howToUse.html>